Information Processing & Management Vol. 25, No. 1, pp. 55–72, 1989
Printed in Great Britain.

0306-4573/89 $3.00 + .00

MODELS FOR RETRIEVAL WITH PROBABILISTIC INDEXING

NORBERT FUHR
TH Darmstadt, Fachbereich Informatik, Karolinenplatz 5, 6100 Darmstadt, West Germany

Abstract—In this article three retrieval models for probabilistic indexing are described along with evaluation results for each. First is the binary independence indexing (BII) model, which is a generalized version of the Maron and Kuhns indexing model. In this model, the indexing weight of a descriptor in a document is an estimate of the probability of relevance of this document with respect to queries using this descriptor. Second is the retrieval-with-probabilistic-indexing (RPI) model, which is suited to different kinds of probabilistic indexing. For that we assume that each indexing scheme has its own concept of "correctness" to which the probabilities relate. In addition to the probabilistic indexing weights, the RPI model provides the possibility of relevance weighting of search terms. A third model that is similar was proposed by Croft some years ago as an extension of the binary independence retrieval model but it can be shown that this model is not based on the probabilistic ranking principle. The probabilistic indexing weights required for any of these models can be provided by an application of the Darmstadt indexing approach (DIA) for indexing with descriptors from a controlled vocabulary. The experimental results show significant improvements over retrieval with binary indexing. Finally, suggestions are made regarding how the DIA can be applied to probabilistic indexing with free text terms.

1. INTRODUCTION

Probabilistic information retrieval models are based on the probabilistic ranking principle, which says that documents should be ranked according to their probability of relevance with respect to the actual request. It can be proved that this principle yields an optimum ranking under certain conditions [1]. In the past, most of the probabilistic models investigated were based on a rather simple document representation, namely binary indexing [2–4]. In this article, we will discuss several models for retrieval with probabilistic indexing and show that significant improvements in retrieval effectiveness can be achieved when binary indexing is replaced by weighted probabilistic indexing.

The first paper on probabilistic indexing was published by Maron and Kuhns [5]. The central idea of their model is to estimate for each descriptor in a document a probability of relevance—the probability that the document is relevant to a request which is formulated with this descriptor. But this model has never been investigated in experiments, because of the problem of estimating the required probabilistic parameters. All suggestions for solving this problem (see also [6]) require too much intellectual effort.

In the meantime, other models of probabilistic, automatic indexing have been developed that are based on certain forms of document representation. The well-known 2-Poisson model [7–9] uses the within-document frequency of terms for the estimation of the indexing weights. The Darmstadt indexing approach (DIA), which has been under development since 1978, at the TH Darmstadt, West Germany, is based on a more detailed document representation (see Section 3). Its probabilistic parameters can be estimated either by comparison with manual indexing or from retrieval results. A similar model was suggested in [10].

In this article, we first give an outline of the DIA. All experiments presented here are based on this probabilistic indexing approach. In the following sections, three models for

A related work by Norbert Fuhr was presented during the Pisa ACM SIGIR meeting September 8–10, 1986, and appeared as "Two models of retrieval with probabilistic indexing" on pages 249–257 in "1986—Conference on Research & Development in Information Retrieval," edited by Fausto Rabitti. This final version was submitted January 19, 1988.

retrieval with probabilistic indexing are described. As a generalization of the Maron and Kuhns model, the binary independence indexing model is proposed. The retrieval-with-probabilistic-indexing (RPI) model is a new model for retrieval with probabilistic indexing that is suited for different kinds of probabilistic indexing. Specifically, the probabilistic parameters derived from manual indexing can be given an interpretation in this model and can be related to the probability of relevance. This model is compared with a similar one proposed by Croft [11].

Our experimental evaluation of the different models is described in Sections 6–8. However, since all our experiments use index terms from a controlled vocabulary, probabilistic indexing for free text terms is still an open problem. In Section 9, we therefore conclude by showing how the basic ideas of the DIA can be applied in this situation to estimate probabilistic index term weights based on a more detailed document representation.

2. THE DARMSTADT INDEXING APPROACH

In this section, a brief description of the DIA is given (for further details, see [12–14]). The DIA is a dictionary-based indexing approach for automatic indexing from document titles and abstracts, with a prescribed indexing vocabulary. In the AIR retrieval test [13] it was demonstrated that the DIA is suited even to broad subject fields such as physics.

The indexing task consists of two steps, a description step and a decision step. In the description step information about the relationship between a descriptor s and the document d to be indexed is collected. This information forms the decision base for the second step, the estimation of the probability that the assignment of s to d would be correct.

The description step uses an indexing dictionary. The main part of the indexing dictionary consists of a weighting function $r(s,t)$, where the $r(s,t)$ approximates the probability $P(C|s,t)$ that the assignment of descriptor s to a document that contains term t would be "correct." Therefore, we regard correctness as an event that plays the same role in the indexing process as does relevance in retrieval (see below). Terms can be single words, noun phrases, or formula identifiers (assigned to formulas by specific algorithms).

The description algorithm starts with the identification of terms in the text. As this task cannot be done perfectly, each term is identified in a certain form of occurrence v, where different forms of occurrence are associated with different levels of confidence (see also Section 9). If a term t is identified in a document d and an entry $r(s,t)$ is stored in the dictionary, a descriptor indication from t to s is generated. It contains

- the form of occurrence v of t in d,
- the entry $r(s,t)$,
- further information about s, t, and d.

The collection of all descriptor indications from a document d leading to the same descriptor s is called the relevance description $y(s,d)$ of s with respect to d.

The decision step uses the relevance description $y = y(s,d)$ to estimate the probability $P(C|y)$ that, if relevance description y is given, the corresponding descriptor assignment would be correct. This estimation is done by the indexing function $a(y)$. Different methods for the development of indexing functions have been investigated for the DIA. In [15] a probabilistic formula for this purpose is described. Here we will concentrate on the polynomial approach developed by Knorz [12,16], which uses polynomial classifiers. For this approach, the relevance description y is mapped to a description vector $\mathbf{y}$. The definition of this mapping has to be done heuristically [12,16]. Then a coefficient vector $\mathbf{a}$ is computed such that $\mathbf{a} \cdot \mathbf{y}$ is an estimate of $P(C|\mathbf{y})$.

The DIA is based on the concept of "correctness." For the construction of the indexing dictionary and the development of the indexing function, learning samples of documents with correct descriptor assignments must be given. Within the DIA, no assumptions are made about the kind of these samples. For pragmatic reasons, manually indexed documents were used for this purpose in the past. Now experiments with indexing functions derived from relevance judgments for retrieval results have been made for the first time.

In this case, the probability $P(C|y(s_i,d_m))$ can be used as an estimate for the probability of the binary independence indexing (BII) model, the probability that document d_m is relevant to a request using descriptor s_i in its query formulation (see next section). Retrieval experiments of this kind are described in Section 8.

3. THE BINARY INDEPENDENCE INDEXING MODEL

The binary independence indexing model, which will be described here, is a generalized version of the Maron and Kuhns model. Every specific user request to a retrieval system must be transformed into a query with descriptors from the set $S = \{s_1, \ldots, s_n\}$. We assume that a query can be represented as a binary vector $\mathbf{x} = (x_1, \ldots, x_n)$ with

$$x_i = \begin{cases} 1, & \text{if the query contains descriptor } s_i \\ 0, & \text{otherwise.} \end{cases}$$

In this way, a specific user request f_k is mapped onto a vector $\mathbf{x}_k$ (different requests may have the same query vector).

The event space of the BII model consists of all document–request relationships between the set of all documents in the collection and all requests to the system. As the set of all requests is not completely known, we assume that we have knowledge about a representative sample of it. A document–request relationship is either relevant or nonrelevant, which will be denoted by R and $\bar{R}$, respectively.

The BII model seeks for an estimate of $P(R|\mathbf{x}_k, d_m)$, the probability that the document d_m is relevant to a request using query $\mathbf{x}_k$. Four versions of BII will be considered below, based on application of the following three independence assumptions:

$$P(\mathbf{x}_k) = \prod_{i=1}^{n} P(x_{k_i}) \tag{1}$$

$$P(\mathbf{x}_k|R, d_m) = \prod_{i=1}^{n} P(x_{k_i}|R, d_m) \tag{2}$$

$$P(\mathbf{x}_k|\bar{R}, d_m) = \prod_{i=1}^{n} P(x_{k_i}|\bar{R}, d_m). \tag{3}$$

All three assumptions relate to the distribution of descriptors in the queries. Formula (1) says that the distribution of the descriptors in all queries is independent, whereas formulas (2)/(3) say that the distribution of the descriptors is independent only in those queries where the document d_m is relevant/nonrelevant to the corresponding request.

Using assumptions (1) and (2), we get the ranking formula BII1:

$$P(R|\mathbf{x}_k, d_m) = P(R|d_m) \cdot \prod_{i=1}^{n} \frac{P(R|x_{k_i}, d_m)}{P(R|d_m)}. \tag{4}$$

Here $P(R|d_m)$ denotes the probability that d_m is relevant to an arbitrary request, and $P(R|x_{k_i}, d_m)$ is the probability that document d_m is relevant to an arbitrary request that contains descriptor s_i in its query ($x_{k_i} = 1$) resp. where s_i is not present in the query ($x_{k_i} = 0$).

With (3) instead of (1), we get the odds formula BII2, where $O(X) = P(X)/P(\bar{X})$:

$$O(R|\mathbf{x}_k, d_m) = O(R|d_m) \cdot \prod_{i=1}^{n} \frac{O(R|x_{k_i}, d_m)}{O(R|d_m)}. \tag{5}$$

In the original model of Maron and Kuhns, the following simplifying assumption was made implicitly:

$$\prod_{\substack{i \\ x_{k_i}=0}} P(x_{k_i}=0) \approx \prod_{\substack{i \\ x_{k_i}=0}} P(x_{k_i}=0|R,d_m) \tag{6}$$

This means that the relevance of a document with respect to a request depends only on those descriptors that are present in the query, and not on those descriptors that the query does not contain.

With (1), (2), and (6) we get the original ranking formula of Maron and Kuhns, which we call BII3:

$$P(R|\mathbf{x}_k,d_m) = P(R|d_m) \cdot \prod_{\substack{i \\ x_{k_i}=1}} \frac{P(R|x_{k_i}=1,d_m)}{P(R|d_m)}. \tag{7}$$

If we use assumption (3) instead of (1) together with (2) and (6), we get the ranking formula BII4:

$$O(R|\mathbf{x}_k,d_m) = O(R|d_m) \cdot \prod_{\substack{i \\ x_{k_i}=0}} \frac{O(R|x_{k_i}=1,d_m)}{O(R|d_m)}. \tag{8}$$

The probabilistic parameters required for an application of the BII model can be estimated on the basis of the DIA (see Section 7). In Section 8, we give experimental results for the different ranking formulas of the BII model in comparison to the other models discussed in the following.

4. A GENERAL MODEL FOR RETRIEVAL WITH PROBABILISTIC INDEXING

The retrieval-with-probabilistic-indexing (RPI) model described here is similar to the so-called 2-Poisson-independence (TPI) model described in [17]. The main difference between the TPI model and the RPI model is that the RPI model is suited to different probabilistic indexing schemes, whereas the TPI model is an extension of the 2-Poisson model for multiterm queries. The TPI model makes use of the specific assumptions of the indexing model, so that for any other indexing model a new retrieval model would have to be developed.

We assume that the event space of the indexing model consists of document-descriptor relationships (ddr), and that a specific ddr is either correct or not. The concept of correctness can be regarded as a pragmatic standard, which differs from one indexing model to another: For most applications of the DIA, manual indexing forms this standard. The model proposed in [10] is also based on manual indexing. In the 2-Poisson model, correctness is replaced by the relevance of the document to all queries containing only the descriptor considered, and in the BII model the relevance to queries containing this descriptor is regarded.

The RPI model deals with request-document relationships, so its central concept is "relevance." To link the two concepts "relevance" and "correctness" together, the RPI model needs additional relevance information about the relationship between a request and the correctness of certain descriptors, that is the relationship between a request and (hypothetical) documents with a certain correct indexing. The event space of the RPI model is—in principle—the same as that of the BII model: all document-request relationships are regarded. In contrast to the BII model, the RPI model is able to distinguish between different requests using the same query formulation. However, as any retrieval system has restricted knowledge about a request, the notation f_k used in the probabilistic formulas below does not relate to a single request but stands for a set of requests about which the system has the same knowledge. Equally, d_m relates to the system's representation of documents.

To give a more explicit definition of the event space of the RPI model, let F denote the set of all requests and D the set of all documents in the collection. Then the event space is the Cartesian product $F \times D$, where each element (document-request pair) has a relevance judgment and the correct indexing of the document (a set of descriptors) associated with it. We can extend this event space from binary to weighted indexing by assuming that we have a fixed number of binary indexings for every document. For example, one could have a group of indexers where each of them has to index every document. Let I be the set of indexers, then the event space is the Cartesian product $F \times D \times I$. Associated with every element of this event space is a relevance judgment and the set of descriptors assigned by the indexer. However, to keep the following explanations simple, we will relate to the first version of the event space.

An event, a document-request pair, is regarded with respect to correct indexings. For that we use the binary vector $\mathbf{x} = (x_1, \ldots, x_n)$ where each component corresponds to an element of the set of descriptors S. Here $x_i = 1$ stands for the event that the correct indexing of a document contains descriptor s_i, and $x_i = 0$ in the contrary case. Using this notation, the probability of correctness estimated by the indexing model can be written as

$$P(x_i = 1 | d_m) = P(C | s_i, d_m).$$

The relevance relationship between a request f_k and the set of documents with a descriptor s_i correctly assigned is described with probabilities of the form $P(R|x_i, f_k)$, that is the probability that a randomly selected document to which s_i was assigned ($x_i = 1$) resp. not assigned ($x_i = 0$) correctly is relevant to request f_k.

We denote the set of all possible correct indexings as X, where $|X| = 2^n$. Now we regard the document-request relationship between d_m and f_k with respect to all indexings $\mathbf{x} \in X$. For the probability of this event we get

$$P(R | f_k, d_m) = \sum_{\mathbf{x} \in X} P(R | \mathbf{x}, f_k) \cdot P(\mathbf{x} | d_m). \tag{9}$$

Now we apply Bayes' theorem:

$$P(R | f_k, d_m) = \sum_{\mathbf{x} \in X} P(R | f_k) \cdot \frac{P(\mathbf{x} | R, f_k)}{P(\mathbf{x} | f_k)} \cdot P(\mathbf{x} | d_m). \tag{10}$$

Equation (10) is a general formula for retrieval with probabilistic indexing. Here all dependencies between descriptors can be considered. Before we apply some independence assumptions to simplify this formula, let us have a short look at the different probabilities involved here: $P(R|f_k)$ is the probability that a (randomly selected) document would be relevant to request f_k. As this probability is constant for one request, there is no need for its estimation when only a ranking of documents for this request is desired. $P(\mathbf{x}|R, f_k)$ is the probability that a relevant document (w.r.t. f_k) has the correct indexing $\mathbf{x}$. The probability $P(\mathbf{x}|f_k) = P(\mathbf{x})$ is independent of the specific request; it stands for the probability that a (randomly selected) document has the correct indexing $\mathbf{x}$. Finally, $P(\mathbf{x}|d_m)$ is the probability that the indexing $\mathbf{x}$ is correct for document d_m.

Now we will make three independence assumptions. As eqn (10) is a general formula for retrieval with probabilistic indexing, it would also be possible to make assumptions that include certain dependencies between descriptors.

$$P(\mathbf{x} | d_m) = \prod_{i=1}^{n} P(x_i | d_m) \tag{11}$$

$$P(\mathbf{x}) = \prod_{i=1}^{n} P(x_i) \tag{12}$$

$$P(\mathbf{x} | R, f_k) = \prod_{i=1}^{n} P(x_i | R, f_k). \tag{13}$$

With (11), we assume that the correctness of a descriptor in a document is independent of the correctness of other descriptors within the same document. This assumption refers to the underlying indexing model; all indexing models mentioned are based on this assumption. Assumption (12) says that the distributions of correct descriptors within the documents are independent of each other. For the relation between relevance and correctness, we assume with (13) that the correctness of a descriptor in a (randomly selected) document that is relevant to a request f_k is independent of the correctness of other descriptors in this document.

With these assumptions, we get from eqn (10):

$$P(R|f_k,d_m) = P(R|f_k) \sum_{\mathbf{x}\in X} \prod_{i=1}^{n} \frac{P(x_i|R,f_k)}{P(x_i)} \cdot P(x_i|d_m). \tag{14}$$

This can be transformed into

$$P(R|f_k,d_m) = P(R|f_k) \cdot \prod_{i=1}^{n} \left(\frac{P(x_i=1|R,f_k)}{P(x_i=1)} \cdot P(x_i=1|d_m) + \frac{P(x_i=0|R,f_k)}{P(x_i=0)} \cdot P(x_i=0|d_m) \right). \tag{15}$$

With the following notations:

$$p_{ik} = P(x_i=1|R,f_k)$$

$$q_i = P(x_i=1)$$

$$u_{im} = P(x_i=1|d_m) = P(C|s_i,d_m)$$

we get

$$P(R|f_k,d_m) = P(R|f_k) \cdot \prod_{i=1}^{n} \left(\frac{p_{ik}}{q_i} u_{im} + \frac{1-p_{ik}}{1-q_i} (1-u_{im}) \right). \tag{16}$$

In this formula, only descriptors s_i with $p_{ik} \neq q_i$ have an influence on the resulting probability $P(R|f_k,d_m)$. Therefore, we can conclude that the query formulation for f_k should exactly contain all these descriptors, which we will denote as set f_k^S. From the point of view of the indexing model, descriptors with a probability $P(C|s_i,d_m) = 0$ are not assigned to a document. Therefore, let d_m^S be the set of descriptors with $u_{im} > 0$.

Using these two sets of descriptors, we can simplify eqn (16), thus getting the ranking formula RPI1:

$$P(R|f_k,d_m) = P(R|f_k) \cdot \prod_{s_i \in f_k^S \cap d_m^S} \left(\frac{p_{ik}}{q_i} u_{im} + \frac{1-p_{ik}}{1-q_i} (1-u_{im}) \right) \cdot \prod_{s_i \in f_k^S \setminus d_m^S} \frac{1-p_{ik}}{1-q_i}. \tag{17}$$

By analogy to Maron and Kuhns' indexing formula, we can make another simplifying assumption: the relevance of a document with respect to a request is not affected by descriptors that occur in the query formulation only but not in the document's indexing:

$$\prod_{s_i \in f_k^S \setminus d_m^S} P(x_i=0|R,f_k) \approx \prod_{s_i \in f_k^S \setminus d_m^S} P(x_i=0). \tag{18}$$

In this case the second product of formula (17) is approximately 1. Therefore, it can be omitted, and we get the ranking formula RPI2:

$$P(R|f_k,d_m) = P(R|f_k) \cdot \prod_{s_i \in f_k^S \cap d_m^S} \left(\frac{p_{ik}}{q_i} u_{im} + \frac{1-p_{ik}}{1-q_i} (1-u_{im}) \right). \tag{19}$$

To apply the ranking formulas RPI1 or RPI2, the parameters p_{ik} and q_i must be estimated in addition to the indexing weights u_{im} which come from the indexing model. The values q_i can be derived from the indexing weights in the document collection D:

$$q_i = P(x_i = 1) \approx \sum_{d_m \in D} \frac{u_{im}}{|D|}. \tag{20}$$

The parameters p_{ik} are request-specific. They can be estimated by using relevance feedback information from a small set of documents D'. Let w_{mk} be the value of the relevance judgment of d_m with respect to f_k with $w_{mk} = 1$ if the document is relevant and $w_{mk} = 0$ otherwise. Then we can use the following estimation:

$$p_{ik} = P(x_i = 1|R,f_k) \approx \frac{\sum_{d_m \in D'} u_{im} w_{mk}}{\sum_{d_m \in D'} w_{mk}}.$$

The formulas given above only show how the parameter estimation should be done in principle. Of course, better estimates are possible. We will discuss this problem in a forthcoming paper.

5. CROFT'S EXTENSION OF THE BINARY INDEPENDENCE RETRIEVAL MODEL

The well-known binary independence retrieval model (BIR) [2,3] is a probabilistic retrieval model suited to a binary indexing of documents. Croft developed an extension of this model for the combination with weighted probabilistic indexing in [11], and evaluated it later [18]. Here we will give a short description of these models and compare Croft's model with the RPI model.

In the BIR model, a document d_m is represented by a binary vector $\mathbf{x}_m = (x_{m_1}, \ldots, x_{m_n})$ where $x_{m_i} = 1(0)$ if the descriptor $s_i \in S = \{s_1, \ldots, s_n\}$ has (not) been assigned to the document. Among the different forms of the BIR model described in [3], we will only regard the most widely used one:

$$O(R|f_k,\mathbf{x}_m) = \frac{P(R|f_k,\mathbf{x}_m)}{P(\bar{R}|f_k,\mathbf{x}_m)} = O(R|f_k) \cdot \frac{P(\mathbf{x}_m|R,f_k)}{P(\mathbf{x}_m|\bar{R},f_k)}.$$

Here odds are used instead of probabilities. $O(R|f_k,\mathbf{x}_m)$ is a monotonic function of $P(R|f_k,\mathbf{x}_m)$, the probability that a document represented by the binary vector $\mathbf{x}_m$ is relevant to the request f_k. As $O(R|f_k)$ is constant for one request, it can be omitted if only a ranking of the documents for the request is needed. For ease of computation, the logarithm of the remaining factor is regarded, which is usually denoted as $g(\mathbf{x})$. (The exact notation should be $g_k(\mathbf{x}_m)$, because $g(\mathbf{x})$ is a specific function for the actual request f_k and $\mathbf{x} = \mathbf{x}(d_m) = \mathbf{x}_m$.) Note that we still have a monotonic function of the probability of relevance $P(R|f_k,\mathbf{x}_m)$.

With the assumptions that the descriptors are distributed independently in all relevant and all nonrelevant documents we get:

$$g(\mathbf{x}) = \log \frac{P(\mathbf{x}_m|R,f_k)}{P(\mathbf{x}_m|\bar{R},f_k)} = \log \prod_{i=1}^{n} \frac{P(x_{m_i}|R,f_k)}{P(x_{m_i}|\bar{R},f_k)}.$$

After doing some simplifications (see e.g. [11]), we end up with

$$g(\mathbf{x}) = \sum_{s_i \in f_k^S} x_{m_i} \cdot \log \frac{p_{ik}(1 - q_{ik})}{q_{ik}(1 - p_{ik})} + \sum_{s_i \in f_k^S} \log \frac{1 - p_{ik}}{1 - q_{ik}}, \tag{21}$$

where $p_{ik} = P(x_{m_i} = 1 | R, f_k)$, $q_{ik} = P(x_{m_i} = 1 | \bar{R}, f_k)$, and $p_{ik} = q_{ik}$ for all $s_i \notin f_k^S$, the set of query terms. The second sum of this function is a constant C_k for a specific query and therefore does not affect the ranking of the documents. So $g(\mathbf{x})$ can be thought of as a simple linear matching function where each search term has a weight

$$\log \frac{p_{ik}(1 - q_{ik})}{q_{ik}(1 - p_{ik})}.$$

The basic idea of Croft's extension to this model is to use the probabilistic index term weights for the computation of the expected value of the above ranking function $g(\mathbf{x})$, ranking documents according to this value $E[g(x)]$.

For this purpose, the binary weights x_{m_i} are replaced by the probabilistic weights $P(x_{m_i} = 1) = u_{im}$, thus getting the ranking function EGX:

$$E[g(\mathbf{x})] = \sum_{s_i \in f_k^S} u_{im} \cdot \log \frac{p_{ik}(1 - q_{ik})}{q_{ik}(1 - p_{ik})} + C_k.$$

In our experiments, we regard two simplified versions of this ranking function (see also Section 7). First, only estimates for the q_{ik}s are given while a global value p for the p_{ik}s is assumed. In this case, we get

$$E[g(\mathbf{x})] = \sum_{s_i \in f_k^S} u_{im} \log \frac{1 - q_{ik}}{q_{ik}} + \sum_{s_i \in f_k^S} u_{im} \log \frac{p}{1 - p} + C_k \tag{22}$$

$$= \sum_{s_i \in f_k^S} u_{im} \log \frac{1 - q_{ik}}{q_{ik}} + C_p \sum_{s_i \in f_k^S} u_{im} + C_k, \tag{23}$$

with $C_p = \log[p/(1 - p)]$. This simplification (already derived in [18]) also shows that the contributions from the term weights p_{ik} and q_{ik} to the value of $E[g(\mathbf{x})]$ can be separated.

Our second application of the EGX formula uses global estimates for the q_{ik}s as well as for the p_{ik}s. For this, we can further simplify eqn (23) to

$$E[g(\mathbf{x})] = C \cdot \sum_{s_i \in f_k^S} u_{im} + C_k, \tag{24}$$

where $C = \log[q/(1 - q)] + \log[p/(1 - p)]$. Here the choice of the constant C (which should be positive, of course) does not affect the ranking of the documents. So the EGX models says that in this case, documents should be ranked according to their sum $\Sigma_{s_i \in f_k^S} u_{im}$.

Although the EGX model uses the same information as the RPI model, its derivation is simpler and the ranking formula is less complex. But this approach has a major deficency: its ranking value $E[g(\mathbf{x})]$ is not a monotonic function of the probability of relevance $P(R | f_k, d_m)$ because of the logarithmic transformation used in $g(\mathbf{x})$. The crucial point is that

$$E\left(\log \frac{P(d_m | R, f_k)}{P(d_m | \bar{R}, f_k)}\right) \neq \log E\left(\frac{P(d_m | R, f_k)}{P(d_m | \bar{R}, f_k)}\right).$$

The fact that the ranking resulting from $E[g(\mathbf{x})]$ is not based on the probability ranking principle can be illustrated by an example. Suppose that we have in a collection D two disjoint sets of documents D_1 and D_2, where the documents within each set have the same representation so that they get the same rank by application of a ranking function based on this representation. For an actual request f_k, a decision has to be made whether to rank set D_1 or D_2 higher. Let us further assume that each set D_i can be divided into two subsets D_{i1} and D_{i2} and that we have knowledge about the distribution of relevant and nonrelevant documents within each subset as shown in Table 1.

Obviously, the documents from D_1 should be given the higher rank: for a randomly selected document from D_1, the probability of relevance (40/80) is higher than that of a document from D_2 (39/79). We get the same ranking from regarding the values of $g(\mathbf{x})$, which are zero for D_1 and negative for D_2. But when we use the knowledge about the subsets for the computation of the values of $E[g(\mathbf{x})]$ for the two sets D_1 and D_2, we get

$$E(g(D_1)) = \frac{70}{80} \cdot g(D_{11}) + \frac{10}{80} \cdot g(D_{12}) = \frac{70}{80} \cdot \log \frac{39/79}{31/80} + \frac{10}{80} \cdot \log \frac{1/79}{9/80} \approx -0.07$$

$$E(g(D_2)) = \frac{69}{79} \cdot g(D_{21}) + \frac{10}{79} \cdot g(D_{22}) = \frac{69}{79} \cdot \log \frac{30/79}{39/80} + \frac{10}{79} \cdot \log \frac{9/79}{1/80} \approx 0.05.$$

This means that the ranking function $E[g(\mathbf{x})]$ does not yield a ranking according to the probability ranking principle!

6. TEST SETTING

For the experiments described in the following, the collection from the AIR retrieval test [13] was taken. As this test used Boolean search formulations (without NOT operators), retrieval was made in two steps. In the first step, conventional Boolean retrieval with queries from the AIR test (which included descriptors only) was performed. For these retrieval runs, a very broad unweighted document indexing was chosen by applying a cutoff-value of 0.01 to the weighted indexing called A1 in [13], which is based on the polynomial approach (in the AIR test a cutoff-value of 0.12 had been used). In the following, we only regard the sets of output documents selected this way. The second retrieval step is performed for every ranking formula considered, thus ranking the documents selected once by the first retrieval step. The queries for the application of the ranking formulas consist of the sets of descriptors from the corresponding Boolean queries.

In the first retrieval step, only 244 from the original 309 queries of the AIR test had nonempty answer sets. These 244 queries were divided randomly into three samples named A, B, and C. Samples B and C were used for parameter adaption only (see next section) and sample A was taken for the ranking experiments. Sample A includes 79 queries that retrieved 2,835 documents altogether from the collection of 14,956 documents; the distribution of answer sizes is shown in Table 2. The relevance judgments of the answer documents are values from the relevance scale listed in Table 3 (in the original test collection, there were some documents judged as "nondecidable," which have been removed from the answer sets regarded here).

Table 1. An example where $E[g(x)]$ gives a ranking different from that of the probabilistic ranking principle

	D_1		D_2	
	D_{11}	D_{12}	D_{21}	D_{22}
R	39	1	30	9
$\bar{R}$	31	9	39	1

Table 2. Distribution of answer sizes in sample A

Retrieved documents per query	Number of queries
1	8
2-5	16
6-10	11
11-20	16
21-50	13
51-100	10
>100	5

Table 3. Relevance scale

Relevant
Conditionally relevant/more relevant
Conditionally relevant
Conditionally relevant/more irrelevant
Irrelevant
Digressive

For evaluation, we use the normalized recall measure as defined in [19] for multistage relevance scales. This measure only considers documents in different ranks and with different relevance judgments. A pair of these documents is in the right order if the document with the higher relevance judgment comes first; otherwise it is in the wrong order. Let S^+ be the number of document pairs in the right order, S^- the number of those in the wrong order, and $S^+_{\max}$ the number of documents in the right order for an optimum ranking. The normalized recall is then defined as follows:

$$R_{\text{norm}} = \frac{1}{2}\left(1 + \frac{S^+ - S^-}{S^+_{\max}}\right).$$

A random ordering of documents will have an R_{norm} value of 0.5 on average. For the cases with $S^+_{\max} = 0$ we defined $R_{\text{norm}} = 1$. Because of the large scattering of the answer sizes, we use a second average method besides the macro average R^M_{norm}: the micro-macro average R^m_{norm} is a weighted average with respect to the answer sizes. Let n_i be the answer size of retrieval result Δ_i, then the micro-macro average of R_{norm} for a set of t queries is defined as:

$$R^m_{\text{norm}}(\Delta_1, \ldots, \Delta_t) = \frac{\sum_{i=1}^{t} n_i \cdot R_{\text{norm}}(\Delta_i)}{\sum_{i=1}^{t} n_i}.$$

For 12 of our 79 queries (which retrieved 43 documents altogether), the value of $S^+_{\max}$ equals 0 and so any ordering of documents will have a value of $R_{\text{norm}} = 1$ according to the definition given above. As a consequence of this, the average R_{norm}-values for random ordering of our test collection are $R^m_{\text{norm}} = 0.505$ and $R^M_{\text{norm}} = 0.576$.

We use the multivalue relevance scale for evaluation because this yields a finer measure of differences in retrieval quality. In [20], it is proved that—under certain conditions—a ranking on the basis of probability of relevance also yields a good ranking according to the degree of relevance. Some experiments not described here have shown that the difference between retrieval results remains the same, whether a binary or a multivalue relevance scale is used for evaluation [21].

7. ESTIMATION OF PROBABILISTIC PARAMETERS

To apply the ranking formulas described above three kinds of probabilistic parameters have to be estimated: (1) weighted indexing for all formulas; (2) for the BII model, request-independent document weights; and (3) search term weights for the RPI and EGX models.

Table 4 shows how the different samples were used for estimation of parameters for each of the models under consideration. Sample X and the AIR collection are disjoint sets of documents, while samples B and C (as well as A) are subsets of the AIR collection. With the parameters, we have denoted the cases where they are estimated for one of the indexings (i.e., A1 or I1 as described below) only or where global values are estimated instead of request-, document- or descriptor-specific ones. The estimation process for the different parameters is described in detail in the following.

We have developed two kinds of probabilistic indexing for the experiments described in the following. Both indexings are based on the DIA as described in Section 3. For the development of the indexing function, the polynomial approach has been applied. The two indexings differ in the definition of correctness to which the probabilistic parameters relate:

- Indexing A1 was taken from the AIR retrieval test. The indexing function was derived from manual indexing [22] using a learning sample of 1,000 documents with about 24,000 relevance descriptions.
- Indexing I1 was adopted on the basis of the retrieval results of the query sample B. According to our application of the BII model, for each descriptor of the query the assignment to a document is assumed to be correct if the document is either judged to be relevant or (one of the three forms of) conditionally relevant on the relevance scale used; otherwise the assignment of this descriptor is false. As there were only 2,822 documents in this sample, this means that there are only 2,822 independent decisions on the basis of which the polynomial approach can be adopted. Therefore, this indexing cannot be optimized as much as A1. The formulas BII1 and BII2 additionally require "absence weights" $P(R|x_i = 0, d_m)$ for descriptors of a document that do not occur in the query. For I1, these indexing weights were estimated in the same way as for the descriptors that were both in the document and in the query.

The request-independent document weights $P(R|d_m)$ also were estimated using the polynomial approach. For this purpose, sample B was used again. The DIA was varied in the way that all descriptor indications from one document formed one "relevance description," which is judged to be correct only if the document is relevant (or conditionally relevant) to the current request. For the search term weights p_{ik} and q_i of the RPI model and p_{ik} and q_{ik} of the EGX model, two kinds of estimates were used:

Table 4. Estimation of probabilistic parameters on the different samples used

		Model		
Sample	Sample size	BII	RPI	EGX
X	1,000	$P(R\|x_{k_i} = 1, d_m)^{A1}$	u_{im}^{A1}	u_{im}^{A1}
AIR coll.	14,956		q_i^{A1}	q_{ik}^{A1}
B	2,822	$P(R\|x_{k_i}, d_m)^{I1}$ $P(R\|d_m)^{I1}$	u_{im}^{I1}	u_{im}^{I1}
C	2,819	$P(R\|d_m)^{global}$	p_{ik}^{global} q_{ik}^{global}	

- In most of the experiments, global estimates for these parameters were applied, that is, all terms have the same parameter values. In contrast to the EGX model [see eqn (24)], different parameter combinations with $p > q$ will yield different ranking results for the RPI model in this case. But here also the estimation of these parameters is not critical. In test runs with varied parameters on sample C we found that different pairs (p,q) lead to nearly equal retrieval results. For the experiments, the values $p = 0.20$ and $q = 0.15$ were selected for both indexings.
- For the ranking formulas denoted with the suffix IDF in the following, inverse document frequencies were used as estimates for parameters q_{ik} and q_i, while again global values were taken for the p_{ik}s, choosing the best values from some test runs on sample C. This approach has been successfully evaluated for the EGX model in [18]. We found that for both models different definitions of the IDF weights (counting the number of documents vs. summing up the index term weights, applying different cutoff values before counting/summing, division by the number of documents in the collection vs. division by the largest term frequency/weight sum) had no influence on the retrieval results, and also any p value in the range of 0.4 . . . 1 gave nearly equal results.

No results of experiments with search term weights based on relevance feedback data are given here because there was no appropriate test sample available (see also [23]).

8. EXPERIMENTS

With the different ranking formulas, experiments were made using sample A of the test collection. The results are given in Table 5. Experiments 1–13 deal with the BII model. In experiment 1 only the document weights were used for ranking, and it can be seen that these document weights are in fact useful for document ranking. But the results of exper-

Table 5. Results of experiments

No.	Formula	Doc. weight	Indexing	R^{M}_{norm}	R^{m}_{norm}
0	Random ord.			0.576	0.505
1	Doc. weight	$P(R\|d_m)$		0.637	0.541
2	BII1	$P(R\|d_m)$	I1	0.569	0.495
3	BII2	$P(R\|d_m)$	I1	0.581	0.493
4	BII3	$P(R\|d_m)$	I1	0.662	0.649
5	BII4	$P(R\|d_m)$	I1	0.671	0.657
6	BII1	0.5	I1	0.657	0.552
7	BII2	0.5	I1	0.657	0.545
8	BII3	0.5	I1	0.733	0.700
9	BII4	0.5	I1	0.732	0.702
10	BII3	0.5	A1	0.732	0.685
11	BII4	0.5	A1	0.716	0.676
12	BII3	0.2	A1	0.740	0.693
13	BII4	0.2	A1	0.751	0.700
14	RPI1		I1	0.726	0.704
15	RPI2		I1	0.729	0.701
16	EGX		I1	0.728	0.700
17	RPI1		A1	0.768	0.735
18	RPI2		A1	0.762	0.721
19	RPI1/IDF		A1	0.690	0.626
20	EGX		A1	0.769	0.733
21	EGX/IDF		A1	0.769	0.731
22	Cosine		I1	0.709	0.701
23	Cosine		I1/bin	0.677	0.628
24	Cosine		A1	0.769	0.731
25	Cosine		A1/bin	0.740	0.688

iments 2–5 show that those estimates of $P(R|d_m)$ are not suitable for the application of the BII model. For experiments 6–9, the document weights estimated as described above were replaced by 0.5, the average value of the estimates of $P(R|d_m)$, and this leads to much better results. At the moment these results cannot be fully explained. We suppose that the document weight $P(R|d_m)$ and the indexing weights $P(R|x_i, d_m)$ are too dependent on each other, so that their combination in one formula (based on the assumption that they are independent) does not work. The other remarkable result from experiments 2–9 is that formulas BII3 and BII4 work significantly better than BII1 and BII2. The difference between these formulas is that BII1 and BII2 also consider the descriptors of a document that do not occur in the query, whereas formulas BII3 and BII4 are restricted to those descriptors that document and query have in common. Other experiments in which only a few descriptors with extremely high or low estimates of $P(R|x_i = 0, d_m)$ were considered did not show any improvement over the results for formulas BII3 and BII4. Therefore, it can be concluded that the "nonasked" descriptors (occurring in the document only) do not have any influence on the probability of relevance. Comparing the results of the probability- (BII1, BII3) and the odds-formulas (BII2, BII4), there is nearly no difference. This means that the Maron and Kuhns formula BII3 is an approach for the probabilistic indexing that cannot be improved by other assumptions of the BII model.

The R_{norm}-values of experiments 10 and 11 show that this approach does not work well with indexing A1: although the indexing quality of A1 is better than that of I1 (see below), the ranking results are worse. The reason for this result is that the probabilistic weights of A1 are related to the concept of correctness derived from manual indexing rather than to the concept of relevance, according to our definition of "relevant" on the multistage relevance scale. (The latter concept is the basis for the global estimate of 0.5 for $P(R|d_m)$.) To find an estimate for $P(R|d_m)$ that is better suited to the indexing weights of A1, we performed some retrieval runs with different values for this parameter on sample C. In experiments 12 and 13, we get slightly improved results with the new choice for $P(R|d_m)$.

The results of the experiments with the RPI model are also listed in Table 5. Obviously, indexing A1 fits better to the RPI model than to the BII model. The sign test shows a significant difference between the results of experiments 13 and 17 at a confidence level of 99%. For indexing I1, nearly the same results as with the BII model are obtained. This demonstrates that the RPI model can be applied to different kinds of probabilistic indexing.

The results of the ranking functions of the RPI and the EGX model for indexing A1 are significantly better (sign test: >99%) than those for I1. As mentioned before, this difference in the indexing quality probably depends on the relatively small learning sample available for the development of I1.

The two RPI functions and the EGX function (without IDF weights) provide nearly identical results. To all appearances, the theoretical deficiency of the EGX model does not have any consequences on its retrieval effectiveness. When IDF weights are used, it performs significantly better than the RPI model. It seems that the RPI model is not suited to the use of IDF weights in combination with global p values. The reason for this might be that the RPI formula cannot be separated in the same way as the EGX formula (23) where the q_{ik}s contribute independently from the p value to the ranking value. On the other hand, the EGX/IDF function does not perform better than the function without IDF weights in our experiments. This result is different from those described in [18] and [24], where significant improvements were gained with the usage of IDF weights. We assume that this is caused by the different kinds of query terms (controlled vocabulary vs. free text terms) and indexing scheme (DIA vs. simple weighting scheme) (see also [23]). In the DIA, all available information about a descriptor is collected in the relevance description and contributes to the estimation of the probabilistic index term weight. Therefore, the IDF weight bears no additional information about the term and thus cannot improve retrieval effectiveness.

In experiments 22–25, the cosine measure was used for document ranking. For the cases with weighted indexing (i.e. 22 and 24), there is no great difference from the results

obtained from probabilistic indexing procedures. Although the application of the cosine measure might be easier, the probabilistic models have the advantage of being more transparent, because the underlying assumptions are made explicit.

To show the benefit of using weighted instead of binary indexing for retrieval, experiments 23 and 25 were made with binary indexings. These indexings were derived by applying optimal cutoff-values (estimated on sample C) to the corresponding weighted indexings. For both indexings, we get significant worse (sign test: >99%) results. This statement also holds for other ranking functions not discussed here: in any case, we get significant improvements of retrieval effectiveness when weighted indexing is used instead of binary indexing.

9. PROBABILISTIC INDEXING WITH FREE TEXT TERMS

Although the experiments described above have shown the superiority of weighted over binary indexing, there is still the problem of estimating the probabilistic index term weights in the case where no controlled vocabulary is used. Over the years, different attempts have been made to derive these weights from the within-document frequencies of the terms (see e.g. [18,25,26]), and significant improvements of retrieval quality were gained. But for the only probabilistic approach to solve this problem, the 2-Poisson model [7–9], no improvement over binary indexing could be shown [17,27]. Obviously, the basic assumptions of the 2-Poisson model are inappropriate.

Here we propose a new approach for the estimation of index term weights that is based on the concept of the form of occurrence (FOC) from the DIA [14]. This concept is more powerful than the approaches mentioned before. The basic idea is that the task of identifying terms in a document cannot be done perfectly. Instead of having a single definition of term occurrence that serves as a basis for the decision whether a specific term is identified in the actual document or not, we allow several such definitions that we call FOC, where different FOCs correspond to different levels of confidence. Actually, the concept of FOC comprises two aspects: (1) the certainty with which a term is identified, and (2) the significance of a term with respect to the document. These two aspects cannot always be separated exactly—there is also no need for it.

In the development work based on the DIA, different parameters for the definition of FOCs have been investigated and shown to be useful. (Only a few attempts have been made to assign explicit weights to specific FOCs, because the concept of the DIA is such that the assignment of weights is postponed until all available information has been gathered in the description step.) For the intended application, the process of assigning a probabilistic weight to a term in a document works as follows: first, the FOC of the term within the document is determined, and then the term is given the weight belonging to this FOC. For the estimation of the FOC weights, the conceptual framework of the RPI model forms a useful guideline: a small learning sample is needed from which the decision about the correctness of (free text) terms w.r.t. documents can be derived. Then for each FOC the ratio of "correct" terms (which we call the precision of the FOC) can be estimated.

In the following subsections, we describe some relevant parameters for the definition of FOCs.

Term class

In the applications based on the DIA, up to three term classes have been distinguished: single words, noun phrases, and formula identifiers. Not only is it appropriate to have quite different FOCs for distinct term classes (see below), for equivalent FOCs of two term classes it is also possible to estimate different precision values. Another criterion for the definition of term classes might be the document frequency of the terms.

Word stemming

For the application of the DIA, two types of word stemming have been used. (In [28], three word stemming algorithms are compared with respect to their influence on retrieval quality. In contrast to our approach, the different stemming algorithms are only used for

a binary identification of terms, so no term weighting (in the form discussed here) is performed.) In addition to the stemming widely used in experimental work in information retrieval, we also regard the "basic form" of a word, which is the infinitive of verbs and the singular of nouns. Table 6 shows some of these basic forms and full word forms for the word stem "comput." Of course, the FOCs based on the basic form have a higher precision value than those using the word stem only.

Distance or grammatical structure of noun phrases

The identification of noun phrases in a text is a difficult task. In our experiments, two different approaches to solve this problem have been shown to be useful. In the first approach, where several formal parameters (e.g. word sequence, sentence boundary) were investigated, only the distance between the first and the last component of the noun phrase in the text provided a useful basis for the distinction of FOCs with different precision values. The exact definition of the distance measure did not have a significant influence on the results. The second approach is based on the grammatical structure of the noun phrases in the document. Therefore, a partial parsing (based on word classes that are a byproduct of the stemming algorithm) of the noun phrases is performed. We found that in this approach we get more significant differences between the precision values for different FOCs than in the approach based on the distance measure. In contrast to the usual application of parsing, where only a binary decision about the occurrence or nonoccurrence of a noun phrase is made, our approach yields more useful information about the certainty of identification.

Location within the document

This seems to be one of the most important parameters for distinguishing FOCs leading to significant differences in precision values. In the experiments based on the DIA, the documents only consisted of titles and abstracts, but the distinction between these two locations proved to be extremely useful (see below). In actual databases, documents have quite a number of parts (e.g. subject headings, controlled terms, classification, journal title) in which a free text term can be identified.

Within-document frequency

In the absence of appropriate models for the distribution of terms within a document, the absolute number of occurrences of the term can be used as a criterion to distinguish different FOCs. Alternatively, one could also compute some ratio (e.g. if there is a great variation in the length of the documents) and distinguish intervals of this ratio. The advantage of this method in comparison to the nonprobabilistic approaches cited above is that in every case we have probabilistic weights for the different FOCs.

To illustrate the last two concepts, let us have a look at the FOCs for single words shown in Table 7. The "location" means that either at least one occurrence of the word

Table 6. Some basic word forms and full word forms of the word stem "comput"

Word stem	Basic word form	Full word form
Comput	Compute	Compute
		Computed
		Computes
		Computing
	Computer	Computer
		Computers
	Computerize	Computerize
		Computerized
	Computerization	Computerization
	Computation	Computation
	Computational	Computational

Table 7. FOCs for single words in their basic form

FOC			
Location	wdf	$P(v)$	$P(C\|v)$
Title	≥1	0.157	0.36
Abstract	≥1	0.489	0.15
Title	≥3	0.053	0.39
Title	=2	0.067	0.38
Title	=1	0.038	0.30
Abstract	≥3	0.030	0.22
Abstract	=2	0.071	0.19
Abstract	=1	0.392	0.15

is in the title (location = title) or that all occurrences are within the abstract (location = abstract), and wdf relates to the within-document-frequency of the word. The precision values are derived from the comparison with manual indexing within the application of the DIA (on a test sample with 24,000 relevance descriptions). $P(v)$ is the probability that the specific FOC v occurs in a random relevance description, and $P(C|v)$ is the probability that such a relevance description leads to a correct descriptor assignment. For example, the first entry in Table 7 reads as follows: 15.7% of all relevance descriptions contain an FOC where the word occurs at least once in the title, and 36% of these relevance descriptions lead to correct descriptor assignments. There is a significant difference between words in the title and those in the abstract, while the within-document frequency has little influence on the precision values $P(C|v)$. The wdf values are relatively small because we regard the basic word forms here.

For an application of the FOC concept, there is still the problem of how to combine the parameters described above (or additional ones) for the definition of FOCs. The crucial point is that all occurrences of a term in a document have to be comprised in a single FOC. Therefore, the development of appropriate FOCs has to be done experimentally. A very simple approach would be to define heuristically several classes of FOCs (as in the example above) and estimate the precision values of each class from a learning sample. But the results from the work with the DIA also can be applied here: one can regard the complete information about the occurrence of a term t in a document d as a relevance description $y = y(t,d)$ and then apply the methods cited in Section 2 for the development of an indexing function $a(y)$.

It is obvious that the definition of the FOCs and the estimation of the weights (the development of the indexing function) depends on the document collection. This is the main advantage of the FOC approach: instead of defining an abstract weighting scheme, the definition of FOCs allows the estimation of the correct probabilities. While the first approach can be verified only indirectly by regarding its retrieval effectiveness, in the FOC approach every probabilistic weight has an explicit notion, and there are theoretical models indicating how these weights should be combined in the retrieval function.

10. CONCLUSIONS

The experimental results described in this article show that probabilistic indexing can successfully be used for ranking procedures, and that significant improvements over retrieval with binary indexing are achieved. The DIA has been used as a basis for the application of different probabilistic indexing models: with the DIA, it is possible to estimate probabilistic indexing weights required for the Maron and Kuhns model, and it has been shown that this model yields good ranking results. Here it should be emphasized that the indexing dictionary that has been used for the development of the I1 indexing (based on retrieval results) was the same as that used for A1, for which both the indexing dictionary and the indexing function were derived from manually indexed documents. At the moment, there is no possibility to build up a dictionary on the basis of retrieval results

because there is not enough data available. But all the ranking results confirm that the estimation of the probabilistic parameters on the basis of manual indexing, the approach that has been chosen for the experiments with the DIA so far, is a practical and successful method for the development of automatic indexing systems.

A major advantage of the BII model has not been mentioned before: in contrast to all other probabilistic indexing and retrieval models, the BII model yields direct estimates of the probability of relevance for a given request–document pair. In other models, there are too many probabilistic parameters that would have to be estimated for the computation of this probability, so only a ranking of the documents is performed. We think that the estimation of the probability of relevance would be a useful feature of a system based on probabilistic retrieval because this information could give the user some notion of the retrieval quality of the documents retrieved.

For the RPI model it has been shown that this model is suited to different kinds of probabilistic indexing. It is more flexible than the BII model because it works with two concepts: "correctness" as a basis of the underling indexing model, and "relevance" for the retrieval parameters. In addition, the RPI model provides the possibility of assigning probabilistic weights to the search terms.

The EGX model developed by Croft is very similar to the RPI model and gives nearly identical experimental results. In contrast to the RPI model, it also can be applied when only IDF weights are available. However, the EGX model is not a probabilistic retrieval model in the original sense because it is not based on the probability ranking principle.

Although all experiments in this article are based on probabilistic indexing with descriptors from a prescribed vocabulary, we think that there is some possibility that the models proposed here might also work successfully with probabilistic indexing for free text terms. The concepts developed within the DIA seem to be easily transferable to this situation, providing two main advantages over comparable approaches: a richer and more detailed document representation and an explicit probabilistic weighting scheme without unrealistic assumptions.

REFERENCES

1. Robertson, S.E. The probability ranking principle in information retrieval. Journal of Documentation, 33: 294–304; 1977.
2. Salton, G.; Yu, C.T. Precision weighting—an effective automatic indexing method. Journal of the Association for Computing Machinery, 23: 76–85; 1976.
3. Robertson, S.E.; Sparck Jones, K. Relevance weighting of search terms. Journal of the American Society for Information Science, 27: 129–146; 1976.
4. van Rijsbergen, C.J.; Harper, D.J. An evaluation of feedback in document retrieval using co-occurrence data. Journal of Documentation, 34: 294–304, 1978.
5. Maron, M.E.; Kuhns, J.L. On relevance, probabilistic indexing, and information retrieval. Journal of the Association for Computing Machinery, 7: 216–244; 1960.
6. Maron, M.E. Probabilistic approaches to the document retrieval problem. In: Salton, G.; Schneider, H.-J., editors. Research and development in information retrieval. Berlin, Heidelberg, New York: Springer; 1982: 98–107.
7. Harter, S.D. Probabilistic approach to automatic keyword indexing, Part I: On the distribution of speciality words in a technical literature. Part II: An algorithm for probabilistic indexing. Journal of the American Society for Information Science, 26: 197–206, 280–289; 1975.
8. Bookstein, A.; Swanson, D.R. Probabilistic models for automatic indexing. Journal of the American Society for Information Science, 25: 312–318; 1974.
9. Bookstein, A.; Swanson, D.R. A decision theoretic foundation for indexing. Journal of the American Society for Information Science, 26: 45–50; 1975.
10. Robertson, S.E.; Harding, P. Probabilistic automatic indexing by learning from human indexers. Journal of Documentation, 40(4): 264–270; 1984.
11. Croft, W.B. Document representation in probabilistic models of information retrieval. Journal of the American Society for Information Science, 32(6): 451–457; 1981.
12. Knorz, G. A decision theory approach to optimal automatic indexing. In: Salton, G.; Schneider, H.-J., editors. Research and development in information retrieval. Berlin, Heidelberg, New York: Springer; 1982: 174–193.
13. Fuhr, N.; Knorz, G. Retrieval test evaluation of a rule based automatic indexing (AIR/PHYS). In: van Rijsbergen, C.J., editor. Research and development in information retrieval. Cambridge: Cambridge University Press; 1984: 391–408.
14. Lustig, G., editor. Automatische Indexierung zwischen Forschung und Anwendung. Hildesheim, Zürich, New York: Georg Olms; 1986.

15. Fuhr, N. A probabilistic model of dictionary-based automatic indexing. RIAO 85, Recherche d'Informations Assistée par Ordinateur; 1985 March 18-20; Grenoble, France; 207-216.
16. Knorz, G. Automatisches Indexieren als Erkennen abstrakter Objekte. Sprache und Information, Band 8. Tübingen: Niemeyer; 1983.
17. Robertson, S.E.; van Rijsbergen, C.J.; Porter, M.F. Probabilistic models of indexing and searching. In: Oddy, R.N.; Robertson, S.E.; van Rijsbergen, C.J.; Williams, P.W., editors. Information retrieval research. London: Butterworths; 1981; 35-56.
18. Croft, W.B. Experiments with representation in a document retrieval system. Information Technology, 2(1): 1-22; 1983.
19. Bollmann, P.; Jochum, R.; Reiner, U.; Weissmann, V.; Zuse, H. Planung und Durchführung der Retrievaltests. In: Schneider, H.-J., et al., editors. Leistungsbewertung von Information Retrieval Verfahren (LIVE). Projektabschlußbericht TU Berlin, Computergestützte Informationssysteme (CIS), Institut für angewandte Informatik, Fachbereich Informatik; 1986: 183-212.
20. Bookstein, A. Outline of a general probabilistic retrieval model. Journal of Documentation, 39(2): 63-72; 1983.
21. Fuhr, H. Probabilistisches Indexing und Retrieval. Dissertation, TH Darmstadt, FB Informatik; 1986.
22. Knorz, G. Development of automatic indexing for the AIR retrieval test. Experiments by means of ALIBABA. Internal Report DVII 83-3, TH Darmstadt, Fachbereich Informatik, Fachgebiet Datenverwaltungssysteme 2; 1983.
23. Fuhr, N.; Müller, P. Probabilistic search term weighting—some negative results. In: van Rijsbergen, C.J.; Yu, C.T., editors. ACM conference on research and development in information retrieval; 1987 June 2-5; New Orleans, USA.
24. Croft, W.B.; Harper, D.J. Using probabilistic models of document retrieval without relevance information. Journal of Documentation, 35: 285-295; 1979.
25. Salton, G.; Yang, C.S.; Yu, C.T. A theory of term importance in automatic text analysis. Journal of the American Society for Information Science, 36: 33-44; 1975.
26. Salton, G. Recent trends in automatic information retrieval. In: Rabitti, F., editor. ACM conference on research and development in information retrieval; 1986 September 8-10; Pisa, Italy.
27. Losee, R.; Bookstein, A.; Yu, C.T. Two Poisson and binary independence assumptions for probabilistic document retrieval. In: Rabitti, F., editor. ACM conference on research and development in information retrieval; 1986 September 8-10; Pisa, Italy.
28. Harman, D. A failure analysis of the limitation of suffixing in an online environment. In: van Rijsbergen, C.J.; Yu, C.T., editors. ACM conference on research and development in information retrieval; 1987 June 2-5; New Orleans, USA.