Information Retrieval, 3, 253–271, 2000

Retrieving Information from a Distributed Heterogeneous Document Collection

CHRISTOPH BAUMGARTEN baumgarten@eurospider.com
Eurospider Information Technology AG, Zurich, Switzerland

Received August 26, 1999; Revised August 26, 1999; Accepted June 12, 2000

Abstract. This paper describes a probabilistic model for optimum information retrieval in a distributed heterogeneous environment.

The model assumes the collection of documents offered by the environment to be partitioned into subcollections. Documents as well as subcollections have to be indexed, where indexing methods using different indexing vocabularies can be employed. A query provided by a user is answered in terms of a ranked list of documents. The model determines a procedure for ranking the documents that stems from the Probability Ranking Principle: For each subcollection, the subcollection's documents are ranked; the resulting ranked lists are combined into a final ranked list of documents, where the ordering is determined by the documents' probabilities of being relevant with respect to the user's query. Various probabilistic ranking methods may be involved in the distributed ranking process. A criterion for effectively limiting the ranking process to a subset of subcollections extends the model.

The property that different ranking methods and indexing vocabularies can be used is important when the subcollections are heterogeneous with respect to their content.

The model's applicability is experimentally confirmed. When exploiting the degrees of freedom provided by the model, experiments showed evidence that the model even outperforms comparable models for the non-distributed case with respect to retrieval effectiveness.

Keywords: distributed information retrieval, heterogeneity, probability ranking principle

1. Introduction

An *information retrieval* (*IR*) system is a tool for searching information in a collection of documents that satisfies a user's information need. In order to use an IR system, the user has to formulate his or her information need; the resulting *query* is then taken by the system to *rank* the documents according to an estimate of their *probabilities of being relevant to the user's information need*. For this purpose, a corresponding *ranking method* is called with the query. It assigns *retrieval status values* (RSVs) to the documents. A document's RSV determines the position of the document in the *ranked list* of documents that represents the answer to the query.

In the following, we concentrate on ranking methods that operate on *statistical information* quantifying certain query and document *features*. The set of features *indexing* a query or document forms the query's or document's *description*. The generation of descriptions is performed by appropriate *indexing methods*. The *indexing vocabulary* correlated to an indexing method provides all the features that may be used by the method for indexing.

Feature and *document frequencies* (Salton and McGill 1983) are examples for statistical information: The feature frequency counts how often a certain feature can be derived from

a specific query or document. The document frequency gives the number of document descriptions containing a certain feature. *Relevance feedback* given by the user on certain documents implies additional statistical information.

In this paper, a *probabilistic model for retrieving information in a distributed document collection* is presented and experimentally evaluated. A document collection is considered to be distributed, if it is partitioned into *subcollections* that are allocated to various *provider* sites. Searching the subcollections by means of IR methodologies can be done in many different ways (Baumgarten 1999a). The *strategy for distributed IR* corresponding to the probabilistic model is the following: The provider site of a subcollection is responsible for indexing the documents in the subcollection using the respective subcollection-specific indexing method. Besides the provider sites, an identified site exists (namely the *broker* site) that is responsible for controlling the ranking process. This process is distributed:

1. First, a *local ranking* is performed in each subcollection at the provider sites. A subcollection-specific ranking method operates on statistical information gathered from a query description and the descriptions of the documents in the respective subcollection.
2. In a second step, the *local ranked* (*document*) *lists* resulting from the first step are combined at the broker site into a *final ranked* (*document*) *list*.

Beside the combination of local ranked lists, the control task of the broker comprises further subtasks, such as *selecting subcollections*, i.e. limiting the process to those subcollections that can be expected to contribute relevant documents to the ranking result (otherwise, processing and communication capacities are wasted), or influencing (biasing) the ranking process at the provider sites.

The potential problems that can arise with approaches following this strategy are numerous (Baumgarten 1999a). The two most important problems are:

– The *subcollection selection problem*: When selecting subcollections for retrieval, other subcollections might be excluded that should have been considered. The resulting decrease in retrieval effectiveness must be kept as small as possible; on the other hand, the extent of the metadata required by the selection process should be reasonable.
– The *subcollection fusion problem*:[1] The local ranked lists derived on the basis of subcollection-related statistical information have to be merged in such way that no decrease in retrieval effectiveness is caused with respect to a comparable non-distributed setting.

There are approaches that follow the outlined strategy while trying to overcome these two problems. A first example is the approach proposed by Voorhees et al. (Voorhees et al. 1994): For evaluating an incoming query, they suggest exploiting relevance feedback information from training queries evaluated before run time, in order to compute (at the broker site) the number of top-ranked documents to be taken from each subcollection. The final ranking is obtained by interleaving the selected documents while preserving the local ranking order of the documents: The more documents are selected from a subcollection, the higher these documents are ranked.

An approach that can also be related to the strategy described above is the one presented by Callan et al. (Callan et al. 1995): They suggest employing an inference network to rank not only documents at the provider sites, but also subcollections at the broker site, and to use the subcollection-ranking for selecting subcollections and for fusing local rankings.

A third example is the further development of the GlOSS approach (Gravano and Garcia-Molina 1995) described by Meng et al. (Meng et al. 1998): Their approach to overcome the subcollection fusion problem shows some similarities to the one presented in this paper and first outlined in Baumgarten (1997). However, the metadata to be maintained at the broker site is more extensive and the agreement on one indexing vocabulary used all over is required. The subcollection fusion problem is solved by either using globally valid document frequencies (if the same cosine-based ranking method is used at each site) or by re-ranking selected documents at the broker site.

Note in this context that the possibility of using different ranking methods and indexing vocabularies at the different sites is an important property when being forced to retrieve information from highly *heterogeneous* subcollections. It provides additional degrees of freedom that can be exploited in order to take subcollection-specific document properties into consideration.

An interesting cost-based selection criterion to solve the subcollection selection problem is suggested by Fuhr (Fuhr 1999); the goal is to receive the maximum number of documents at minimum cost. This criterion requires the estimated number of relevant documents within a subcollection as well as an approximation of the recall-precision graph correlated to a subcollection, both with respect to the current query.

The solutions offered by most of these approaches to overcome the subcollection selection/fusion problem are (at least in parts) heuristic in nature. To our knowledge, no theoretically well founded framework for distributed retrieval is known so far that *integrates* acceptable non-heuristic solutions to the two problems. The probabilistic model described in the following may be considered to be a proposal for such a framework. Moreover, it allows the assignment of different ranking methods and indexing vocabularies to the subcollections.

2. A probabilistic approach for ranking documents in a distributed environment

Probabilistic IR models, as for example the binary independence retrieval model (BIR) (Robertson and Sparck-Jones 1976, Schäuble 1997) for non-distributed IR, are based on the *probability ranking principle* (PRP): Presenting the documents to the user in decreasing order of their probabilities of being relevant with respect to the user's query is optimal (Robertson 1977). To obtain the optimal order of the documents, an *order-preserving transformation* of the probabilities of relevance has to be estimated. The estimation of the probabilities themselves is not required, as we are interested in the ordering of the documents and not in actual probability values.

In the following, documents and queries are interpreted as *events*, which occur with a certain probability. Two different document events are always disjunctive; the same holds for different query events. The probability of relevance of a document d_i with respect to a certain query q can now be denoted as the *conditional probability* $P(R \mid d_i, q)$. Note that

we use the following abbreviations (Schäuble 1997):

$$\begin{aligned} E(M, N) &:= E(M) \cap E(N) \\ P(M) &:= P(E(M)) \\ P(M \mid N) &:= P(E(M) \mid E(N)), \end{aligned}$$

where $E(\cdot)$ denotes an event and M and N event identifiers. Thus, $P(R \mid d_i, q)$ may be read as the probability of the event $E(R)$ of being relevant given that both events $E(d_i)$ and $E(q)$ occur together. (The notation used in this paper is summarized in Table 1.)

We define

$$f(x) := \ln\left(\frac{x}{1-x}\right)$$

and

$$g(q) := \ln\left(\frac{P(\bar{R} \mid q)}{P(R \mid q)}\right)$$

and obtain with

$$\begin{aligned} RSV(d_i, q) &:= f(P(R \mid d_i, q)) + g(q) \\ &= \ln\left(\frac{P(d_i \mid R, q)}{P(d_i \mid \bar{R}, q)}\right) \end{aligned} \tag{1}$$

an order-preserving query-dependent transformation of $P(R \mid d_i, q)$, where ln denotes the *natural logarithm* and $\bar{R}$ the event *not* $E(R)$.

In order to model a distributed document collection, we consider the given *collection of documents* to be partitioned into *disjunctive subcollections*. That means, we have a collection $\boldsymbol{D}$ of subcollections D_j, where each D_j contains documents d_i. $\boldsymbol{D}$ and the D_j are also interpreted as events:

$$\begin{aligned} E(D_j) &:= \bigcup_{d_i \in D_j} E(d_i) \\ E(\boldsymbol{D}) &:= \bigcup_{D_j \in \boldsymbol{D}} E(D_j). \end{aligned}$$

According to Bayes' theorem,

$$P(d_i \mid R, q) = P(d_i \mid D_j, R, q) P(D_j \mid R, q)\,, d_i \in D_j\,.$$

Hence, (1) can be transformed into

$$RSV(d_i, q) = \ln\left(\frac{P(d_i \mid D_j, R, q)}{P(d_i \mid D_j, \bar{R}, q)}\right) + \ln\left(\frac{P(D_j \mid \boldsymbol{D}, R, q)}{P(D_j \mid \boldsymbol{D}, \bar{R}, q)}\right), \tag{2}$$

where $d_i \in D_j$ and $D_j \in \boldsymbol{D}$.

Table 1. Notations used in this paper.

Symbol	Meaning
d_i	Document
D_j	Subcollection
$\boldsymbol{D}$	Collection of subcollections
q	Query
E	Event
P	Probability
$RSV(d_i, q)$	Retrieval status value of d_i w.r.t. q
$f,\ g$	Order-preserving transformations
φ	Indexing feature
$\Phi_j/\boldsymbol{\Phi}$	Indexing vocabulary correlated to $D_j/\boldsymbol{D}$
$\Phi_j(d_i)/\boldsymbol{\Phi}(D_j)$	Description of d_i/D_j, subset of $\Phi_j/\boldsymbol{\Phi}$
$\Phi^i(q)$, $\boldsymbol{\Phi}(q)$	Descriptions of query q
ff	Feature frequency
df	Document frequency
sf	Subcollection frequency
v_j, w_j, v, w	Parameters provided by the RPI model
$\boldsymbol{l}$	Desired length of the final ranked list
l_j	Length of the ranked list computed from D_j
$\boldsymbol{x}$	Variable in the selection criterion
r_j	Random variable modelling the probabilities of relevance of the documents in D_j
R_j	Random variable modelling the RSVs of the documents in D_j
$T_{j,\varphi}$	Random variable modelling the influence of φ on the RSV of a document in D_j
A_j	Distribution of R_j
P_j	Probability density corresponding to A_j
$\mathcal{G}$	Set of shifted gamma distributions
ν, η	Variables (shifted gamma distribution)
μ	Expectation value of a random variable
σ^2	Variance of a random variable
min	Minimum value that can be taken by a random variable
$\tilde{\cdot}$	Approximation of $\cdot$ (example: $\tilde{\mu}$ denotes the estimated expectation value of a random variable)

The addends in Eq. (2) can be estimated on the basis of D_j- and $\boldsymbol{D}$-wide statistical information with the help of probabilistic IR models that have been designed for the non-distributed case, i.e. that give a framework for estimating expression (1). While the first addend results from locally ranking the documents at the provider sites, the the computation of the second addend is part of the control task performed by the broker and requires the subcollections to be indexed (see (Baumgarten 1999a) for a framework for indexing subcollections). The fact that D^i- and $\boldsymbol{D}$-wide statistical information is employed allows us assigning individual indexing vocabularies Φ_j and $\boldsymbol{\Phi}$ to the different D_j and to $\boldsymbol{D}$, respectively.

It is obvious that Eq. (2) could have been further generalized, if the collection of documents would have been partitioned *hierarchically* into disjunctive subcollections. A subcollection at a certain layer of the subcollection hierarchy would then either contain subordinated subcollections or, if the layer under consideration is the bottom layer, documents. Such a generalization provides the key for *scaling* with arbitrary sizes of the document collection. For sake of simplicity, we go on with a three-layers hierarchy (d, D, $\boldsymbol{D}$); Baumgarten (1999a) describes the entire probabilistic model for a n-layers hierarchy; see also Baumgarten (1997).

In Baumgarten (1999a), the procedure for estimating the addends in Eq. (2) is shown by example for the BIR mentioned above as well as the *retrieval-with-probabilistic-indexing* (RPI) model (Fuhr 1992).

When using the BIR model, we obtain

$$\ln\left(\frac{P(d_i \mid D_j, R, q)}{P(d_i \mid D_j, \bar{R}, q)}\right) = \sum_{\varphi \in \Phi_j(q) \cap \Phi_j(d_i)} \ln\left(\frac{P(\varphi \mid D_j, R, q)}{P(\varphi \mid D_j, \bar{R}, q)}\right) + \sum_{\varphi \in \Phi_j(q) - \Phi_j(d_i)} \ln\left(\frac{P(\bar{\varphi} \mid D_j, R, q)}{P(\bar{\varphi} \mid D_j, \bar{R}, q)}\right) \tag{3}$$

for the first addend in Eq. (2), where φ denotes a feature taken from the query description $\Phi_j(q) \subseteq \Phi_j$ corresponding to the respective subcollection D_j. Applying the method of *Bayesian estimates* (see e.g. Fuhr 1993) while assuming no relevance feedback to be available, we obtain

$$\frac{P(\varphi \mid D_j, R, q)}{P(\varphi \mid D_j, \bar{R}, q)} \approx \frac{|D_j| + 1}{2\,\mathrm{df}(D_j, \varphi) + 1} \tag{4}$$

and

$$\frac{P(\bar{\varphi} \mid D_j, R, q)}{P(\bar{\varphi} \mid D_j, \bar{R}, q)} \approx \frac{|D_j| + 1}{2\,(|D_j| - \mathrm{df}(D_j, \varphi)) + 1} \tag{5}$$

as estimates for the fractions arising in Eq. (3), where $\mathrm{df}(D_j, \varphi)$ denotes the document frequency of feature φ with respect to subcollection D_j. The case that relevance feedback is given is handled in Baumgarten (1999a).

The estimation of the second addend in Eq. (2) can be done analogously on the basis of the BIR model: We simply replace d_i by D_j, D_j by $\boldsymbol{D}$, and Φ_j by $\boldsymbol{\Phi}$. Instead of the document

frequency, the subcollection frequency sf of a feature (counting the subcollections being indexed by the fearure) has to be used.

To give a short example illustrating the probabilistic model for distributed IR in combination with the BIR model, let $\boldsymbol{D}$ contains 100 subcollections. We consider two of them, namely D_1 and D_2. Let q be a query that is indexed by a single query feature φ, i.e. $\Phi(q) = \{\varphi\}$. D_1 is one of 15 subcollections that are indexed by this feature ($\mathrm{sf}(\boldsymbol{D}, \varphi) = 15$). D_2 on the other hand is not indexed by φ. According to (4) and (5), we obtain

$$\ln\left(\frac{P(D_1 \mid \boldsymbol{D}, R, q)}{P(D_1 \mid \boldsymbol{D}, \bar{R}, q)}\right) = \ln\left(\frac{P(\varphi \mid \boldsymbol{D}, R, q)}{P(\varphi \mid \boldsymbol{D}, \bar{R}, q)}\right) \approx 1.18$$

or

$$\ln\left(\frac{P(D_2 \mid \boldsymbol{D}, R, q)}{P(D_2 \mid \boldsymbol{D}, \bar{R}, q)}\right) = \ln\left(\frac{P(\bar{\varphi} \mid \boldsymbol{D}, R, q)}{P(\bar{\varphi} \mid \boldsymbol{D}, \bar{R}, q)}\right) \approx -0.53,$$

respectively, for the second addend in Eq. (2). Let the documents d_1 and d_2 be element of D_1 and D_2, respectively. Even if both documents have exactly the same first addend, e.g.

$$\ln\left(\frac{P(d_1 \mid D_1, R, q)}{P(d_1 \mid D_1, \bar{R}, q)}\right) = \ln\left(\frac{P(d_2 \mid D_2, R, q)}{P(d_2 \mid D_2, \bar{R}, q)}\right) \approx -0.75,$$

we obtain different RSVs for them due to the weighting of the subcollections:

$$RSV(d_1, q) \approx -0.75 + 1.18 = 0.43$$
$$RSV(d_2, q) \approx -0.75 - 0.53 = -1.28.$$

When employing the RPI model for estimating the first addend in Eq. (2), we obtain

$$\ln\left(\frac{P(d_i \mid D_j, R, q)}{P(d_i \mid D_j, \bar{R}, q)}\right) = \sum_{\varphi \in \Phi_j(q)} \ln\left(\frac{\frac{P(\varphi \mid d_i)}{P(\varphi \mid D_j)} P(\varphi \mid D_j, R, q) + \frac{P(\bar{\varphi} \mid d_i)}{P(\bar{\varphi} \mid D_j)} P(\bar{\varphi} \mid D_j, R, q)}{\frac{P(\varphi \mid d_i)}{P(\varphi \mid D_j)} P(\varphi \mid D_j, \bar{R}, q) + \frac{P(\bar{\varphi} \mid d_i)}{P(\bar{\varphi} \mid D_j)} P(\bar{\varphi} \mid D_j, \bar{R}, q)}\right). \quad (6)$$

In this equation, four probabilities are arising, namely the probability $P(\varphi \mid d_i)$ for document d_i being indexed by feature φ, the probability $P(\varphi \mid D_j)$ for an arbitrary document in subcollection D_j being indexed by φ, and the probabilities $P(\varphi \mid D_j, R, q)$ and $P(\varphi \mid D_j, \bar{R}, q)$ for an arbitrary relevant or non-relevant document in subcollection D_j being indexed by φ, respectively. Given that no relevance feedback is available, we may estimate these

probabilities by

$$P(\varphi \mid d_i) \approx \begin{cases} \dfrac{\mathrm{ff}(d_i, \varphi)}{\max_{\varphi \in \Phi_j(d_i)}(\mathrm{ff}(d_i, \varphi))} & \text{if } \mathrm{ff}(d_i, \varphi) \neq 0 \\ 0 & \text{else} \end{cases},$$

$$P(\varphi \mid D_j) \approx \frac{1}{|D_j|} \sum_{d_i \in D_j} P(\varphi \mid d_i), \tag{7}$$

$$P(\varphi \mid D_j, R, q) \approx \frac{v_j}{v_j + w_j},$$

and

$$P(\varphi \mid D_j, \bar{R}, q) \approx \frac{v_j + \sum_{d_i \in D_j} P(\varphi \mid d_i)}{v_j + w_j + |D_j|},$$

where $\mathrm{ff}(d_i, \varphi)$ denotes the feature frequency of feature φ in the description $\Phi_j(d_i)$ of document $d_i \in D_j$ (Baumgarten 1999a) handles the case that relevance feedback is given). Other methods for approximating these probabilities could have been used instead. Furthermore, we set $w_j := 1 - v_j$. The parameters $v_j > 0$ remain to be defined—in our experiments described in Section 4, we chose them manually.

The estimation of the second addend in Eq. (2) can be done analogously on the basis of the RPI model: We simply replace d_i by D_j, D_j by $\boldsymbol{D}$, Φ_j by $\boldsymbol{\Phi}$, v_j by $\boldsymbol{v}$ and w_j by $\boldsymbol{w}$ in Eq. (6) and the subsequently given estimations and set $\boldsymbol{w} := 1 - \boldsymbol{v}$. Parameter $\boldsymbol{v}$ remains to be defined and thus, is chosen manually in our experiments.

3. Selection of subcollections

The model described so far enables the complete ranking of a distributed document collection. However, a user is usually only interested in the first part of the ranked list, namely in the $\boldsymbol{l}$ top-ranked documents. Hence, in order not to waste capacities, the ranking process should be limited to those subcollections D_j, which contain the top-ranked documents. This selection of subcollections is performed at the broker site and thus, should be based on metadata which can be *efficiently* provided to the broker. As a consequence, excluding subcollections or parts of the subcollections from the ranking process might also lead to an exclusion of documents that should have been considered. In other words, the final ranked list of length $\boldsymbol{l}$ will be suboptimal in the sense of the PRP; compare (French et al. 1998). However, since our selection criterion described in the following is properly derived from the PRP, we may say that the occurring corruption is minimized.

3.1. The selection criterion

Starting with a *random experiment*, we randomly pick documents d_i from a certain subcollection D_j and consider their probabilities of relevance. This experiment can be modelled

through the *random variable*

$$r_j : \{d_i \mid d_i \in D_j\} \to [0; 1];$$
$$d_i \mapsto P(R \mid d_i, q)$$

corresponding to D_j and depending on q. Transforming r_j with the functions f and g, we obtain with

$$R_j := f(r_j) + g(q)$$

a second random variable modelling the RSVs of the documents in D_j. Approximating the *discrete distributions A_j of the random variables R_j* correlated to the subcollections D_j, the resulting estimated RSV distributions can be used in order to decide whether (and if yes, how many) documents are taken from a certain D_j.

Assuming a large number of documents to be accessible through D_j, we may approximate R_j's actual distribution A_j by a continuous function

$$\tilde{A}_j(x) := \int_{-\infty}^{x} \tilde{P}_j(t)\, dt, \quad x \in \mathbb{R},$$

the *estimated RSV distribution* of R_j. $\tilde{P}_j$ denotes $\tilde{A}_j$'s *probability density*.

Suppose now that we have decided for distribution functions $\tilde{A}_j$ approximating the actual distributions A_j of the RSVs occurring in the different subcollections D_j. In order to define the *selection criterion* for selecting some D_j from the superior $\boldsymbol{D}$ to be included into the ranking process, we assume the *desired length $\boldsymbol{l}$ of the final ranked list* as given by the user. Determining the variable $\boldsymbol{x} \in \mathbb{R}$ with

$$\boldsymbol{l} = \sum_{D_j \in \boldsymbol{D}} l_j$$
$$l_j := \lfloor |D_j| \, (1 - \tilde{A}_j(\boldsymbol{x})) \rfloor$$

numerically, we obtain for each D_j the number of documents l_j to be taken from this subcollection. Note that the required number $|D_j|$ of documents in D_j can be easily provided at the broker site. If a certain $l_j = 0$, the corresponding D_j is *not* taken into consideration by the retrieval process.

3.2. Approximating the RSV distribution in a subcollection

The estimated RSV distribution $\tilde{A}_j$ can be characterized by a *distribution family* and a *set of statistics*, which uniquely determine a member of the chosen distribution family. If we select these statistics to be approximations of the statistics' corresponding equivalents of the distribution A_j, then the distribution determined by the statistics approximates A_j, provided that the chosen distribution family is suitable in general for modelling RSV distributions.

A possible choice for such a distribution family is the set $\mathcal{G}$ of gamma distributions shifted on the x-axis by a certain smallest possible value (see (Baumgarten 1999b) or (Baumgarten 1999a) for a more general discussion on choosing the distribution family). A *shifted gamma distribution*

$$\tilde{A}_j := \mathcal{G}(\tilde{\mu}(R_j), \tilde{\sigma}^2(R_j), \widetilde{\min}(R_j))$$

as a member of this family approximates the distribution A_j of the random variable R_j modelling the RSVs of the documents in D_j and is determined by its probability density

$$\tilde{P}_j(x) := \begin{cases} \frac{\left(\frac{x-\widetilde{\min}(R_j)}{\nu}\right)^{\eta-1}}{\nu\Gamma(\eta)} e^{\frac{\widetilde{\min}(R_j)-x}{\nu}} & \text{if } x \geq \widetilde{\min}(R_j) \\ 0 & \text{else,} \end{cases}$$

where

$$\eta := \frac{(\widetilde{\mu}(R_j) - \widetilde{\min}(R_j))^2}{\widetilde{\sigma}^2(R_j)},$$
$$\nu := \frac{\widetilde{\sigma}^2(R_j)}{\widetilde{\mu}(R_j) - \widetilde{\min}(R_j)},$$

and Γ denotes the *Gamma function* (Stahel 1995). That means, $\tilde{P}_j$ is uniquely determined by three statistics, namely $\widetilde{\mu}(R_j) = \eta\nu + \widetilde{\min}(R_j)$ as the *expectation value*, $\widetilde{\sigma}^2(R_j) = \eta\nu^2$ as the *variance*, and $\widetilde{\min}(R_j)$ as the *smallest possible value* related to the shifted gamma distribution $\tilde{A}_j$, where $\widetilde{\mu}(R_j)$ is an approximation of R_j's expectation value $\mu(R_j)$, $\widetilde{\sigma}^2(R_j)$ is an approximation of R_j's variance $\sigma^2(R_j)$, and $\widetilde{\min}(R_j)$ is an approximation of R_j's minimum RSV.

We now address the approximation of the statistics at the broker site: First, an approximation $\widetilde{\mu}(R_j)$ of the expectation value $\mu(R_j)$ is developed:

Considering the first addend in Eq. (2), this addend is estimated with the help of a certain existing probabilistic IR model for the non-distributed case. The BIR and RPI models exemplarily used in this paper further transform the addend into a sum over all query features and then estimate values for the resulting feature-related addends; compare Eq. (3) and (6). In fact, most of the known non-distributed probabilistic retrieval models propose a RSV computation that is based on an accumulation over all query features. We will therefore assume for the following, that *the first addend in Eq.* (2) *decomposes into query-feature-related addends*.

The random variable R_j can be rewritten as

$$R_j = R_j^{local} + RSV(D_j, q),$$

where $RSV(D_j, q)$ stands for the value of the second addend in Eq. (2) for subcollection D_j and R_j^{local} is a random variable, which represents the distribution of the *local RSVs* of the documents in D_j, i.e. RSVs that are valid only within the scope of D_j. That is, it models the

influence of the first addend in Eq. (2) on the globally valid RSVs of the documents in D_j. An approximation of $\mu(R_j)$ is obtained, if we find a way to approximate the expectation value $\mu(R_j^{local})$ of R_j^{local}, as

$$\mu(R_j) = \mu\left(R_j^{local}\right) + RSV(D_j, q) \tag{8}$$

due to the linearity of the expectation value.

Due to our assumption made two paragraphs before, R_j^{local} may be conceived as a sum of random variables

$$\begin{aligned} T_{j,\varphi} : \{d_i \mid d_i \in D_j)\} &\longrightarrow \mathbb{R}; \\ d_i &\longmapsto W_j(\varphi, d_i) \end{aligned}$$

describing the *influence of the query features* $\varphi \in \Phi_j$ on a document's RSV:

$$R_j^{local} = \sum_{\varphi \in \Phi_j(q)} T_{j,\varphi}.$$

Here, the function W_j yields the weight of a certain feature $\varphi \in \Phi_j$ with respect to the document $d_i \in D_j$. For sake of simplicity, we assume no relevance feedback data to be available; the case that relevance feedback is given is handled in Baumgarten (1999a). As a consequence, φ's weight is independent of query q. We may therefore consider the corresponding $T_{j,\varphi}$ to be query-independent as well.

As the expectation value of the sum of (not necessarily independent) random variables equals the sum of the random variables' expectation values, we may compute the expectation value of R_j^{local} by

$$\mu\left(R_j^{local}\right) = \sum_{\varphi \in \Phi_j(q)} \mu(T_{j,\varphi})\,.$$

Replacing

$$\mu(T_{j,\varphi}) = \sum_{d_i \in D_j} P(d_i \mid D_j) W_j(\varphi, d_i)$$

by the expectation value

$$\mu(\mu(T_{j,*})) = \frac{\sum_{\varphi \in \Phi_j} P(\varphi \mid D_j)\mu(T_{j,\varphi})}{\sum_{\varphi \in \Phi_j} P(\varphi \mid D_j)} \tag{9}$$

of the random variable $\mu(T_{j,*})^2$ modelling the occurrence of different $\mu(T_{j,\varphi})$, we obtain with

$$\tilde{\mu}\left(R_j^{local}\right) := |\Phi_j(q)|\mu(\mu(T_{j,*})) \tag{10}$$

an estimate for the expectation value of R_j^{local}.

The reasonability of this approximation is linked to the assumption underlying Eq. (9) that the probability for feature φ being a query feature indexing the actual query q is $\frac{P(\varphi \mid D_j)}{\sum_{\varphi' \in \Phi_j} P(\varphi' \mid D_j)}$.

Replacing $\mu(R_j^{local})$ in (8) by the above defined $\tilde{\mu}(R_j^{local})$ finally yields

$$\tilde{\mu}(R_j) := \tilde{\mu}\left(R_j^{local}\right) + RSV(D_j, q). \tag{11}$$

This approximation of $\mu(R_j)$ results from combining D_j's RSV with certain items that rely on the size of the query description $\Phi_j(q)$ as well as further query-independent information, which can be efficiently provided at the broker site:

- The computation of $\mu(\mu(T_{j,*}))$ is built on a before-runtime-determination of the expected influences of the features in Φ_j on the RSV computation. For this purpose, the involved probabilities $P(d_i \mid D_j)$ and $P(\varphi \mid D_j)$ have to be replaced by appropriate estimates. We suggest approximating $P(d_i \mid D_j)$ by $\frac{1}{|D_j|}$ and $P(\varphi \mid D_j)$ by (7).
- $|\Phi_j(q)|$ apriori not only depends on query q, but also on the indexing method used for indexing the documents in D_j. However, in order to avoid the necessity of keeping all the indexing methods available at the broker site, $|\Phi_j(q)|$ should be approximated from the known $|\Phi(q)|$.

In order to derive an approximation $\widetilde{\sigma}^2(R_j)$ for R_j's variance $\sigma^2(R_j)$ and an approximation $\widetilde{\min}(R_j)$ for R_j's minimum RSV $\min(R_j)$, the same fundamental idea that is underlying the approximation of R_j^{local}'s expectation value shown above can be used: Considering the variance of the influence, or the smallest possible influence of those query features on R_j^{local} that play a role in the local RSV computation, we are able to draw conclusions about the variance or minimum RSV of R_j, respectively.

However, considering the variance, things become a bit more complicated: It is a known fact that the occurrence of an indexing feature in the description of a document usually depends on the occurrence of other features in the description (Schäuble 1997). Hence, the random variables $T_{j,\varphi}$ may not be assumed to be independent. While the expectation values of depending random variables add to the expectation value of the sum of these random variables, the variance of a sum of depending random variables also requires the adding of the *covariances* between the random variables to the sum of the random variables' variances. Thus, the approximation of $\tilde{\sigma}^2(R_j)$ includes an adding of the expected covariance between two arbitrary random variables $T_{j,\varphi}$ and $T_{j,\varphi'}$ (Baumgarten 1999a). A before-runtime-computation of this expected covariance in analogy to $\mu(\mu(T_{j,*}))$ would have to take each possible feature-feature pair in Φ_j into account. Such a time intensive procedure can be easily avoided by making use of the results of queries that are assigned to D_j during runtime; see Baumgarten (1999a) for details.

Figure 3 shows a sample RSV distribution A_j that is approximated by a shifted gamma distribution $\tilde{A}_j$ according to the approach described in this section.

4. Experimental evaluation of the model

In the following, the probabilistic model for distributed IR is experimentally evaluated with respect to the *retrieval effectiveness*. The computation of the addends in Eq. (2) is done

on the basis of the RPI model. Hence, we will refer to this variant as the *distributed RPI approach*.

Two common measures of retrieval effectiveness are *recall* and *precision* (Salton and McGill 1983). Considering a ranked list down to a certain depth, the recall gives the percentage of the relevant documents found, while the precision measures the percentage of relevant documents among the documents that were considered.

In order to evaluate our probabilistic model, the *TREC test environment* (Harman 1993) is used. This environment includes a number of large document collections, various sets of topics and corresponding relevance judgements given on documents with respect to a certain topic. We have employed two document (sub)collections from TREC, the *AP88 document collection* containing 79923 news paper articles from the AP Newswire Journal of the year 1988, and the *APFR88 document collection* containing AP88 plus FR88, a collection with 19860 government communications from the US-American Federal Register of the year 1988 (in the following, AP88 and APFR88 will be represented by $\boldsymbol{D}_{AP88}$ and $\boldsymbol{D}_{APFR88}$, respectively). From the content point of view, we may say that AP88 tends to be a rather homogeneous collection, while APFR88 may be considered to be rather heterogeneous.[3]

In order to evaluate the distributed RPI approach, AP88 is partitioned into 81 subcollections $D_j \in \boldsymbol{D}_{AP88}$, $j = 0, \ldots, 80$: AP88 is available on the TREC disk 2 as a set of 322 files, where documents from one file are all from the same date, neighbouring files cover successive time periods. Each four successive files are combined into one subcollection, such that documents are clustered according to their date of appearance. Subcollection D_{80} contains the last two remaining files. The partitioning of APFR88 extends the partitioning of AP88 by 38 subcollections $D_j \in \boldsymbol{D}_{APFR88}$, $j = 81, \ldots, 118$, of more or less equal extend: FR88 is available on the TREC disk 2 as a set of 152 files, where documents from one file are all from the same day, neighbouring files cover successive days. Again, each four successive files are combined into one subcollection.

Different query sets are used for evaluation that are generated from two *sets of topics*, namely *TREC topics 101–150* and *151–200*. Our experiments are performed with *short* as well as *long queries*.[4] The parameter selection is done by manually "optimizing" the results for short queries 101–150. We may therefore consider these queries to be the training queries used for experiments.

Query, document and subcollection descriptions are *automatically generated*: Queries and documents are indexed by eliminating stop words and then applying *Porter's stemming algorithm* (Porter 1980). In order to build a subcollection description, we follow the suggestion in Callan et al. (1995) and employ the union of the descriptions of the documents in the subcollection. The subcollection's feature frequencies result from adding the corresponding feature frequencies correlated to the subcollection's documents. Thus, for indexing documents and subcollections, the same indexing vocabulary is used. However, we do not exploit this fact in our experiments.

First, we compare the retrieval effectiveness of the distributed RPI approach without applying the selection criterion to the retrieval effectiveness measured for the corresponding *non-distributed RPI approach* (in the non-distributed setting, RSVs are computed analogously to Eq. (6) with parameters v and $w := 1 - v$).

– **AP88:** For the experiments on the basis of document collection AP88, we select $v_j = 0.2$ for all j (thus, we abandon the possibility of selecting different values for different v_j),

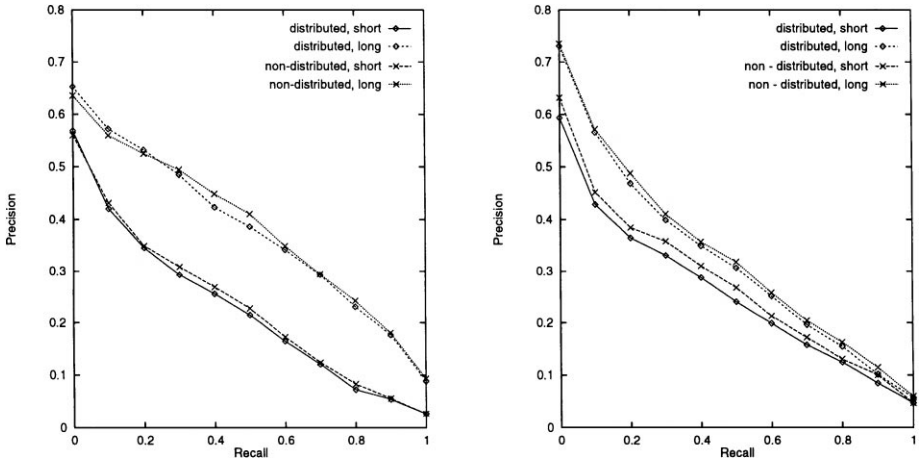

Figure 1. Recall-precision graphs for queries 101–150 (left) and 151–200 (right); document collection AP88.

$\boldsymbol{v} = 0.4$ and $v = 0.2$ (Baumgarten 1999a). To compare the distributed and the non-distributed RPI approach, we consider corresponding *recall-precision graphs* (Salton and McGill 1983) illustrated in figure 1. Obviously, for each query set, the graphs' differences are more or less negligible. This observation is also confirmed by a comparison of *average precisions* (Salton and McGill 1983) in Table 2. Thus, the experiments do not contradict the model's basic prediction: Since the model stems from the PRP, a retrieval effectiveness comparable to the one of a corresponding non-distributed setting should be achieved.

– **APFR88:** Turning towards document collection APFR88, we select $v_j = 0.2$ for $j = 0, \ldots, 80$ (AP part), $v_j = 0.05$ for $j = 81, \ldots, 118$ (FR part), $\boldsymbol{v} = 0.2$ and $v = 0.1$ (Baumgarten 1999a). A comparison of the recall-precision graphs resulting from the distributed and non-distributed RPI approach show clear differences for most of our query sets: For smaller recall values, the various graph progressions corresponding to the distributed case clearly surpass their non-distributed counterparts; see figure 2. This is conspicuous in particular for long queries 151–200, where the difference in early precision is 0.256. Considering the differences in average precision, the upper

Table 2. Comparison of average precisions (AP88).

Query set	Non-distributed RPI approach	Distributed RPI approach
Short, 101–150	0.2221	0.2147
Short, 151–200	0.2578	0.2403
Long, 101–150	0.3751	0.3688
Long, 151–200	0.3162	0.3073

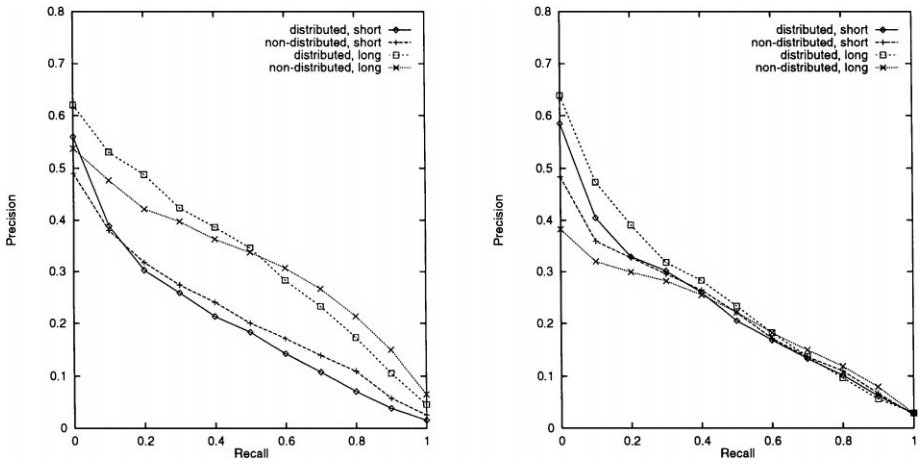

Figure 2. Recall-precision graphs for queries 101–150 (left) and 151–200 (right); document collection APFR88.

hand of the distributed RPI approach continues, if we disregard the results for short queries 101–150; compare Table 3. These experiments show evidence that by using the degrees of freedom provided by the probabilistic model for distributed IR, corresponding nondistributed models can be even outperformed with respect to retrieval effectiveness.

Second, the (inevitable) loss in retrieval effectiveness that occurs when using the distributed RPI approach in combination with the selection criterion is investigated. On the basis of document collection AP88, various runs are performed for different desired lengths $\boldsymbol{l}$ of the final ranked lists, namely for $\boldsymbol{l} = 1000$, 130, 80 and 30. According to the discussion in Section 3.2, we define the $\tilde{A}_j$ approximating the D_j-specific RSV distributions to be shifted gamma distributions and estimate the required statistics as explained.

When performing the distributed RPI approach without the selection criterion, a maximum of 1000 top rated documents are retrieved from each subcollection and merged according to their RSVs. The first 1000 top ranked documents of the resulting ranked list are used as the final result. On the other hand, when making use of the selection criterion, only the first l_j top ranked documents are taken from a subcollection D_j. Thus, even if we select

Table 3. Comparison of average precisions (APFR88).

Query set	Non-distributed RPI approach	Distributed RPI approach
Short, 101–150	0.2029	0.1898
Short, 151–200	0.2074	0.2132
Long, 101–150	0.3063	0.3195
Long, 151–200	0.1971	0.2403

$l = 1000$, much less documents are involved in the building of the final ranked list than without using the selection criterion.

In order to be able to investigate the retrieval effectiveness of the distributed RPI approach with and without selecting subcollections, we measure the precision reached after having checked the first s top ranked documents; see Tables 4 and 5 (due to space limitations, we only report the experimental results for short and long queries 101–150; the results for short and long queries 151–200 are comparable).

For $l > 30$, the decrease in precision ranges between negligible and acceptable; it becomes significant only for s close to l. If $l = 30$, the losses in precision are obvious.

On the other hand, the smaller l has been chosen, the bigger is the average number of subcollections skipped by the selection criterion from a total of 81 subcollections; compare Table 6. Obviously, for $l = 1000$, the selection criterion was more or less not able to eliminate subcollections from the retrieval process. The average number of skipped documents is increasing for $l = 130$ and $l = 80$ and reaches large values for $l = 30$. The longer

Table 4. Precision with/without selection criterion: Short queries 101–150 (AP88).

Precision at s doc.s	No selection	$l = 1000$	$l = 130$	$l = 80$	$l = 30$
5	0.3800	0.3800	0.3680	0.3560	0.3160
15	0.2827	0.2827	0.2800	0.2693	0.2307
30	0.2273	0.2273	0.2220	0.2120	0.1527
50	0.1892	0.1892	0.1788	0.1672	
80	0.1528	0.1528	0.1452	0.1260	
100	0.1394	0.1394	0.1276		
130	0.1243	0.1243	0.1065		
500	0.0545	0.0546			
1000	0.0335	0.0330			

Table 5. Precision with/without selection criterion: Long queries 101–150 (AP88).

Precision at s doc.s	No selection	$l = 1000$	$l = 130$	$l = 80$	$l = 30$
5	0.4600	0.4600	0.4600	0.4600	0.4400
15	0.4147	0.4147	0.4133	0.4040	0.3720
30	0.3640	0.3640	0.3553	0.3393	0.2700
50	0.3144	0.3144	0.3012	0.2864	
80	0.2667	0.2667	0.2480	0.2118	
100	0.2394	0.2394	0.2186		
130	0.2122	0.2120	0.1800		
500	0.0836	0.0835			
1000	0.0454	0.0447			

Table 6. Average number of skipped subcollections (AP88).

Query set	$l = 1000$	$l = 130$	$l = 80$	$l = 30$
Short, 101–150	0	3.9	14.6	52.6
Short, 151–200	0.32	7.2	17.86	52.98
Long, 101–150	0.04	15.96	30.86	58.86
Long, 151–200	0.02	9.42	23.02	54.28

the queries are in average, the more subcollections are skipped (except for $l = 1000$). The reason why for $l > 30$ the average number of skipped subcollections is not larger, is due to the fact that in our setting, the top ranked documents are often distributed over many subcollections. However, for some queries the selection criterion skips much more documents than implied by the average.

Figure 3 shows a randomly picked sample situation that occurred during the application of the selection criterion, where a RSV distribution A_j in a specific subcollection implied by a particular query has been approximated by a shifted gamma distribution $\tilde{A}_j$. Shown is also the error plot illustrating the deviation $\tilde{A}_j(x) - A_j(x)$ for all possible x.

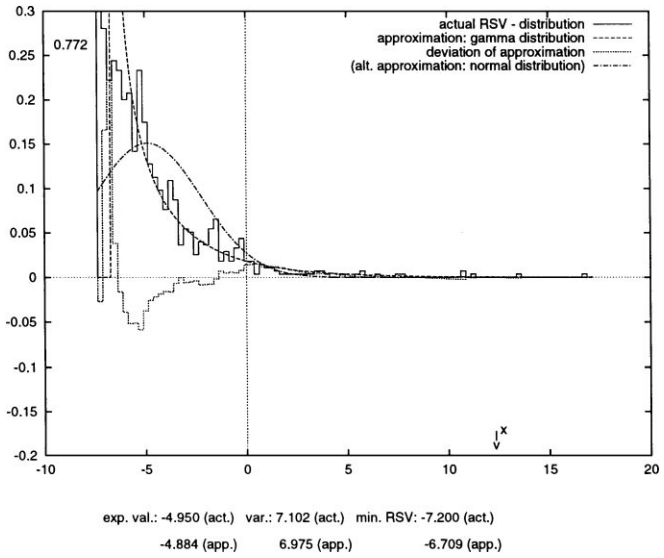

Figure 3. Density of the actual RSV distribution of subcollection D_{53}^1 containing 1098 documents w.r.t. long query 185 indexed by 29 query features; density of the approximating shifted gamma distribution; corresponding error plot. Shown are also the actual and approximated expectation value, variance and minimum RSV. The selection criterion integrates up to x and decides to take one document from the subcollection ($l = 130$). The density value for the minimum RSV is 0.772. Approximating the RSV distributions alternatively by using the set of normal distributions as the distribution family, worse results are obtained, as illustrated by this example.

5. Conclusion and outlook

In this paper, a probabilistic model for distributed IR has been presented that provides a number of desirable properties:

1. The model stems from the Probability Ranking Principle. As a consequence, the achievable retrieval effectiveness is not inferior to the retrieval effectiveness of a corresponding non-distributed model, which has been confirmed by experiments.
2. As an important feature to handle heterogeneity among subcollections, different indexing vocabularies and probabilistic ranking methods may be employed within the framework of the model.
3. The model is able to consider relevance feedback provided by the user, not only given on documents, but also on subcollections.
4. Part of the model is a criterion for limiting the distributed ranking process to a subset of available subcollections. The inevitable losses in retrieval effectiveness caused by the application of the criterion remain acceptable. This has been confirmed by experiments.

In Baumgarten (1999a), various extensions of the presented model are discussed: consideration of cost factors by the selection criterion; generalization of the model such that it can be scaled with the underlying data volume; support for handling (semi-)structured documents.

An interesting point remains for future work: Many degrees of freedom provided by the presented model are not available with non-distributed models. Although our experiments on the basis of document collection APFR88 have exploited only a few of them, the results were encouraging (the distributed RPI approach was in parts able to outperform its non-distributed counterpart). Therefore, further experiments should be performed in order to form an impression of the model's entire potential.

Acknowledgments

This work has been carried out at the former Document and Information Processing Group at the Swiss Federal Institute of Technology in Zurich as well as at the graduate college "Tools for the Effective Use of Parallel and Distributed Computing Systems" at Dresden University of Technology (funded by the German Research Council and the Freistaat Sachsen). The author would like to thank Norbert Fuhr, Klaus Meyer-Wegener, Elke Mittendorf and Peter Schäuble for the numerous discussions and hints.

Notes

1. This problem has been identified by Voorhees et al. (1994) as the "collection fusion problem".
2. μ in combination with the *set* $T_{j,*}$ of all $T_{j,\varphi}$ does not denote an expectation value, but a *random variable* modelling the expectation values corresponding to $T_{j,*}$.
3. Most of the documents in AP88 are relatively small and cover a specific topic. Documents from FR88 in general are much longer and often handle different topics at the same time.
4. A topic consists of fields. Short queries: description field; long queries: description, narrative, summary, concept, factor and definition field (topics 101–150), or description and narrative field (topics 151–200).

References

Baumgarten C (1997) A probabilistic model for distributed information retrieval. In: *Proc. 20th ACM SIGIR Conf. on Research and Development in Information Retrieval*.

Baumgarten C (1999a) Probabilistic information retrieval in a distributed heterogeneous environment. PhD Thesis, Dresden Univ. of Techn.

Baumgarten C (1999b) A probabilistic solution to the selection and fusion problem in distributed information retrieval. In: *Proc. 22nd Annual Int. ACM SIGIR Conf. on Research and Development in Information Retrieval*, Berkeley, CA.

Callan JP, Zhihong L and Croft WB (1995) Searching distributed collections with inference networks. In: *Proc. 18th ACM SIGIR Conf. on Research and Development in Information Retrieval*.

French J, Powell A, Viles C, Emmitt T and Prey K (1998) Evaluating database selection techniques: A testbed and experiment. In: *Proc. 21st ACM SIGIR Conf. on Research and Development in Information Retrieval*.

Fuhr N (1992) Integration of probabilistic fact and text retrieval. In: *Proc. 15th ACM SIGIR Conf. on Research and Development in Information Retrieval*.

Fuhr N (1993) Information retrieval. Course material of the course held in the summer term 1993. University of Dortmund. Available at http://ls6.informatik.uni-dortmund.de/ir/teaching/courses/ir/.

Fuhr N (1999) A decision-theoretic approach to database selection in networked IR. ACM Transactions on Information Systems, 17(3).

Gravano L and Garcia-Molina H (1995) Generalizing GlOSS to vector-space databases and broker hierarchies. In: *Proc. 21st VLDB Conf.*

Harman D (1993) Overview of the first TREC conference. In: *Proc. 16th ACM SIGIR Conf. on Research and Development in Information Retrieval*.

Meng W, Liu K, Yu C, Wang X, Chang Y and Rishe N (1998) Determining text databases to search in the internet. In: *Proc. 24th VLDB Conf.* Extended version.

Porter M (1980) An algorithm for suffix stripping. Program 14.

Robertson S (1977) The probability ranking principle in IR. J. of Documentation, 33(4).

Robertson S and Sparck-Jones K (1976) Relevance weighting of search terms'. J. American Society for Inf. Science, 27.

Salton G and McGill MJ (1983) Introduction to Modern Information Retrieval, McGraw-Hill, New York.

Schäuble P (1997) Multimedia Information Retrieval—Content-Based Information Retrieval from Large Text and Audio Databases, Kluwer Academic, Boston.

Stahel WA (1995) Statistische Datenanalyse, Vieweg Verlag, Braunschweig.

Voorhees EM, Gupta NK and Johnson-Laird B (1994) The collection fusion problem. In: Harman DK (Ed.), Proc. TREC-3.