A RE-UNIFICATION OF TWO COMPETING MODELS FOR DOCUMENT RETRIEVAL

David Bodoff
Department of Information Systems
Leonard N. Stern School of Business
New York University
44 West 4th Street, Suite 9-181
New York, NY 10012-1126
(212) 998-0822
fax: (212) 995-4228
dbodoff@stern.nyu.edu

June 1997

Working Paper Series
Stern #IS-97-10

A Re-Unification of Two Competing Models for Document Retrieval

David Bodoff
Information Systems Department
Stern School of Business
New York University
dbodoff@rnd.stern.nyu.edu
(212) 998 - 0822

Abstract: *Two competing approaches for document retrieval were first identified by Robertson et al (Robertson, Maron et al. 1982) for probabilistic retrieval. We point out the corresponding two competing approaches for the Vector Space Model. In both the probabilistic and Vector Space models, only one of the two competing approaches has received significant research attention, because of the unavailibility of sufficient data to implement the second approach. Because it is now feasible to collect vast amounts of feedback data from users, both approaches are now possible. We therefore re-visit the question of a unification of both approaches, for both probabilistic and Vector Space models. This unification of approaches differs from that originally proposed in (Robertson, Maron et al. 1982), and offers unique advantages. Preliminary results of a simulation experiment are reported, and an outline is provided of an ongoing field study.*

Introduction

Before the advent of the World Wide Web (WWW), computer searches for online documents were limited to isolated stand-alone systems, such as Lexis/Nexis or the MEDLARS medical database. Even then, much research as well as development effort went into improving the success of search results, so that users would more readily find what they were looking for. This area of research is known as information retrieval (IR). With the advent of the WWW, the problem of information retrieval is more pressing, for a number of reasons. First, many more users are searching for online documents. Second, the documents being searched are of very uneven quality and relate to differing topics. Consequently, WWW search engine results are not satisfactory to many users. Third, and perhaps most importantly, intranets are quickly becoming the standard for corporate communication. The importance of corporate knowledge dissemination requires effective search capabilities. Government agencies, such as the US Patents Office and the Securities and Exchange Commission (SEC), are using online databases to disseminate information to the public. In spite of the growing importance of text retrieval, very few studies have measured the effectiveness of retrieval systems in the field. In summary, searching online documents is commonplace and important, but its

effectiveness in realistic settings is uncertain and under-studied.

The same technology which is responsible for the new importance of information retrieval, also offers new possibilities for research. The large number of executed searches represents a rich and easily recorded history. From this history, performance can be measured. Perhaps more importantly, performance may be improved on the basis of those histories. An area of information retrieval research called *relevance feedback* focuses on the automatic improvement of a retrieval system's effectiveness on the basis of this history. We refer to this automatic improvement as "relevance feedback learning".

In this study, we introduce the previous theoretical and experimental work on relevance feedback learning in information retrieval (Sections 1 and 2), and analyze the strengths and shortcomings of that work. A significant observation here is that the *experimental* work has been largely confined to the laboratory, and that one of the two theories of feedback learning has received almost no experimental attention at all. In section three, we argue that the two competing theories of feedback learning have not been fully *theoretically* integrated, in spite of some well-known efforts in that direction. We therefore (1) propose a new unification of the competing theories of feedback learning (Sections 4 and 5), (2) present preliminary results of a test of this new theory (Section 6), and (3) outline a field study (Section 7), currently under way, to establish a baseline of performance for retrieval systems under realistic conditions, and to test the various theories of feedback learning under those realistic conditions.

Part One: Describing the Models

Information retrieval (IR) is "the computer selection of a subset of a document database to display...to a user, usually in response to a user request" (Lewis 1992, February). The retrieval systems we consider here also *rank* the selected documents in order of their predicted relevance to the request. Performance of these systems is measured by the relative rankings of relevant versus irrelevant documents. In the following sections we outline the most popular approaches to information retrieval.

1.1 Vector Space Model

In Salton's Vector Space Model (VSM) (Salton 1989), a document is represented as an n-dimensional vector $D_i = D_{i1},D_{i2},...D_{in}$). Each element D_{ij} of D_i represents the presence or absence of term j in document D_i, or the weight of term j in document D_i. The most common term weightings resemble the well-known tf*idf weights of Luhn (Salton, Yang et al. 1975, January-February; Rijsbergen 1975; Salton and Buckley 1988). A user query is also represented as a vector in the same n dimensions, according to the (weight of) terms present or absent in the query. The document and query vectors are

ordinarily normalized to unit length (Salton 1989). A similarity measure $f_{i,j}$: $D_i \times Q_j \rightarrow \Re$ (Rijsbergen 1975) then ranks documents according to the similarity of the document vector to the query vector, and documents are retrieved to the user in that order. The most common similarity measure is the Cosine coefficient, which in the case of unit-length queries and documents, is simply the inner product of the two vectors. In this way, the VSM is concerned with the question "which documents are close to the query?"

1.2 The Probabilistic Models

The probabilistic models are concerned with estimating the probability of relevance of each document to a query (Maron and Kuhns 1960). These models ask the question "which documents are most probably relevant to the query". The well-known Probability Ranking Principle (Robertson 1977, December) suggests that documents should be presented to the user in decreasing order of their probability of relevance as estimated from the available data.

Many alternatives exist for estimating these probabilities. The family of binary models which we consider in this work (Robertson, Maron et al. 1982) estimates these probabilities by accounting for the presence or absence of query terms in each document. Other approaches such as the Poisson models (Bookstien and Swanson 1974, September/October; Yu, Luk et al. 1979; Harter 1975; Harter 1975) account as well for the *frequency* with which the query terms appear in a document. Still others take a more abstract approach (Fuhr and Buckley 1991, July; Cooper, Chen et al. 1995) which accounts for *properties* of query terms in each document (e.g. does the query term appear in the document title?). In this paper, we focus on the binary models, and leave for future work an extension of the ideas presented here to other probabilistic models.

Robertson et al (Robertson, Maron et al. 1982) carefully delineate four versions of the binary probabilistic models, named Models 0,1,2, and 3. The differences between these models relate directly to our analysis below of transient and permanent learning. Due to lack of space, we will present these models informally, with an example.

Example

The data used by all four models is in the same form as tables one and two below. Table one shows the presence/absence of each term in each document and query, while table two shows a binary assessment of the relevance of documents to queries (many values of this latter table may be unknown): Taken together, these two tables define a joint distribution of document terms, query terms and relevance.

Table 1: Document and Query Contents

	term 1	term 2	term 3
Doc1	1	1	0
Doc2	1	0	1
Doc3	0	0	1
Query1 (Q1)	1	0	0
Query2 (Q2)	1	0	1
Query3 (Q3)	1	1	1

Table 2: Relevance Assessments

	Q1	Q2	Q3
Doc1	1	0	1
Doc2	0	1	1
Doc3	1	1	1

Robertson et al.'s Model 2, sometimes called Binary Independent Retrieval (BIR) (Fuhr 1989), finds the probability of relevance of an *arbitrary* document to a *particular* query Q_l. BIR derives term weights W_{Qlm} for each term m in Q_l. These term weights are computed on the basis of contingency tables which show the reliability of term m in distinguishing documents which are relevant or irrelevant to this query. One intuitively appealing interpretation of these weights is the extent to which the term correctly represents the intended meaning of the query. As an example, the following contingency table is derived from the above data for *Term1 of query Q1*:

Table 3: for Term1 of query Q_l

	Relevant	non-Relevant
Document contains term 1	1	1
Doc. does not contain term 1	1	0

The weight W_{Q11} of term one in query one, would be derived from this table. If $p=P(D_{i1}=1|\text{Relevance}, Q_l)$, meaning the probability that a document contains term 1 given its relevance to query Q_l, $q=P(D_{i1}=1|\text{non-Relevance},Q_l)$, then $W_{Q11}=\log[p*(1-q)/q*(1-p)]$. In this case, the weight would not be high, since not all relevant documents contained term 1, and one non-relevant document did contain it.

How is this weight used? When processing query Q_l, any document containing term m has W_{Qlm} added to its score. The score of each document Di with respect to query Q_l is thus $\Sigma D_{im}*W_{Qlm}$. BIR has been described as calculating the probability of relevance of an arbitrary document to a particular, because its term weights are derived from contingency tables built for a particular query Q_l, and they are applied to all (arbitrary) documents. *BIR thus adopts a particular-query marginal view of the joint data.*

In order to complete the example, we present the exactly analogous approach of Model 1, sometimes called Binary Independent Indexing (BII) (Fuhr 1989). BII estimates the probability of relevance of a particular document to an arbitrary query. In BII, *document* term weights are derived (as opposed to BIR, where query term weights are derived). Using the same joint data as was used for BIR, the contingency table below would allow derivation of the BII weight of *term one in document one*. BII thus adopts a particular-document marginal view of the data.

Table 4: for Term1 of Document D_1

	Relevant	non-Relevant
Qj1=1	2	1
Qj1=0	0	0

After clearly delineating the alternative Models 1 and 2, Robertson et al introduce their Model 0 and Model 3 as the two alternative unifications of Models 1 and 2. Their analysis has served for decades as a framework for understanding the probabilistic models. However, we propose in this work another unification of Models 1 and 2, which has additional advantages. We therefore briefly review their unification models.

Robertson et al's Model 0 estimates the probability that an arbitrary document containing term D_{im} is relevant to an arbitrary query containing term Q_{jm}. This means constructing either query-marginal or document-marginal contingency tables as previously described, except considering all queries or documents together in one table. The other unified model, model 3, integrates information from models 0,1, and 2, to estimate a probability of relevance of a particular document to a particular query. We will return below to discuss this model as it relates to the proposal put forward in this paper.

1.3 A Note on Representation

In this paper, we assume that queries and documents are represented by the words they contain. There is good reason to believe that this naive approach can be improved upon, such as by a factor analysis (e.g. Singular Value Decomposition) which can severely reduce the number of dimensions (Bartell, Cottrell et al. 1992), or by using n-gram (Cavnar 1993) or other representations. For convenience, however, we will speak of each dimension as if it represents a single term in the vocabulary.

1.4 Unifying Framework

The two primary approaches to information retrieval are reflected in these two questions for VSM and the probabilistic approach respectively: "Which documents are closest to the query?", "Which documents are most probably relevant to the query?".

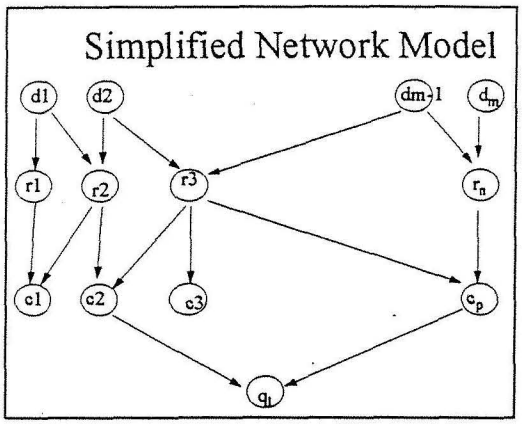
In spite of these different orientations, each approach ultimately specifies a function from documents and queries to a score, denoted above as $f_{i,j}$: $D_i \times Q_j \rightarrow \Re$. Moreover, all these functions can be defined in terms of a small set of transparently meaningful parameters. We adopt a modified version of Turtle and Croft's Inference Network (Turtle 1991) (figure 1) as a common framework into which all retrieval models are positioned.

Each node in the top layer represents a document, each node in the next layer represents a document term, followed by query terms, and a single bottom node representing a single query. Links from a document to a document term represent the presence and strength of that term in the document. A similar meaning holds for links from query terms to the query. The meaning of the middle layer of links is not discussed in this paper.

The VSM and probabilistic models all represent document and query term weights, and define simple scoring functions using those weights. The term weights are easily identified with links in the network, while a scoring function can be identified with the "activation" or local distribution function at the query node. This unifying view is helpful in understanding the common strengths and weaknesses of previous

approaches to feedback learning, the main subject of this paper.

Figure 1 (adapted from (Turtle 1991))



2.0 Learning

Relevance feedback takes the form of a set of triples (D_i,Q_j,R), where R is a binary judgment of the relevance of document D_i to query Q_j.

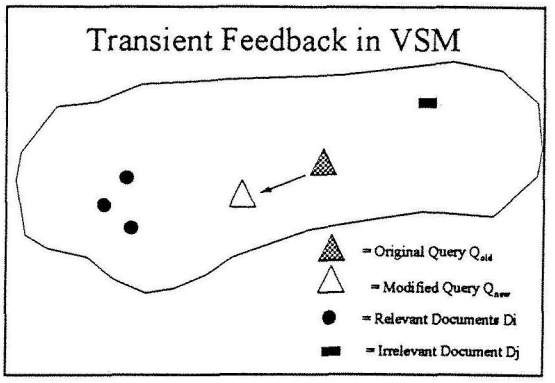
The vast majority of previous research into learning aims to improve the current user's query results, on the basis of the current user's feedback (Harper and Rijsbergen 1978, September; Robertson and Jones 1976, May-June; Salton 1990). For example, all of text categorization research ("routing" in TREC) (Harman, 1992, November; Lewis 1992, February; Masand, Linoff et al. 1992) regards learning to improve results for a given query or topic. These approaches do not allow the system to learn from one user query to the next. We call this *transient feedback*. A smaller amount of research has been done on a different sort of relevance feedback, in which the system learns over time, across user queries (Fuhr 1989; Fuhr and Buckley 1991, July; Cooper, Grey et al. 1992; Kwok 1989; Belew 1989). This sort of relevance feedback, which we call *permanent learning*, is the focus of our work.

2.1 Learning in VSM

In the context of VSM, transient feedback is viewed as re-positioning the query vector in the vector space (Salton 1989). The user submits his initial query, peruses the results, marks the documents he found relevant, then re-submits his query for a second round. Before processing the query in the second round, the system automatically modifies it to include -- or to give additional weight to -- those terms which are found in the documents judged relevant by the user in the first round. This *transient feedback* has the effect of 'moving' the query vector closer to those documents which the user has identified as definitely relevant. Moving the query closer to those known relevant documents has the effect of retrieving, in the second iteration, other similar documents. In a comparison performed in (Salton 1990) the best reweighting function was $Q_{new} = Q_{old} + \Sigma D_i - D_j$, with i ranging over relevant documents and j representing the top-ranked irrelevant document.

In terms of the network of figure 1, this model modifies the weights of links from query terms cm to the query node Q_i, denoted W_{Qicm}. We call this approach transient feedback, because the effects of this learning pertain only to the

Figure 2 (adapted from (Salton 1989))

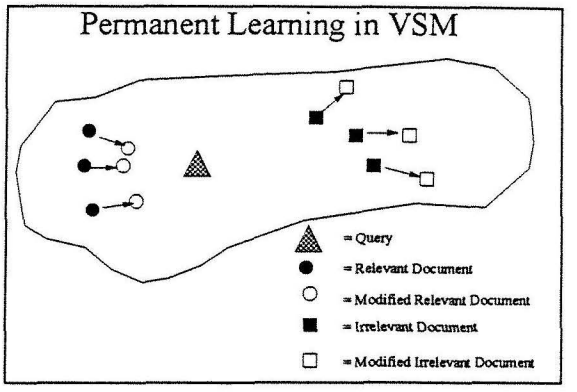


current query, and are removed when a new query is presented. Experimental results have reported very significant benefits of this technique -- relative performance increases on the order of 50-100%, with even one round of relevance feedback (Harman 1992). However, the learning does not benefit subsequent users, even if they present to the system a similar or identical query.

Permanent learning in VSM is viewed as re-positioning the *document* vectors in vector space. This is achieved in a manner very similar to that described above for query modification. In particular, the document vector of relevant documents is modified to include -- or give additional weight to -- those terms which appeared in the query, according to a formula such as $Di_{new} = Di_{old} + \alpha(Ql - Di_{old})$ (Brauen 1971). Irrelevant documents may also be moved away from the query.

In terms of the network model, VSM permanent learning modifies weights from the document nodes to the document term nodes, W_{tmDi}. Promising experimental results were initially reported for this approach, as well. However, subsequent work did not pursue this approach, perhaps due to the lack of sufficient data regarding each individual document.

Figure3



2.2 Learning in Probabilistic Models

In the probabilistic models, the arrival of feedback data is automatically accounted for. As cells in the relevance table (table two above) are filled in, the various contingency tables are updated, along with the derived term weights. Use of this feedback data for transient feedback amounts to using the BIR approach to document retrieval, in which the modifications to the relevance table are important for their effect on the query-oriented contingency tables for the current query. In terms of the network model, updates to the BIR query term weights are viewed as modifying the query term weights W_{Qlcm}, which are re-set with the presentation of each new query. Variations of BIR (Model 2) have been extensively tested and refined, and compared with VSM transient feedback (Harper

and Rijsbergen 1978, September; Harman 1992; Salton 1990; Robertson and Jones 1976, May-June). Improvements are again on the order of 50 - 100% after just one round of feedback.

Use of feedback data for permanent learning amounts to using the BII approach to document retrieval, in which the modifications to the relevance table are important for their effect on the document-oriented contingency tables. In terms of the network model, updates to the BII query term weights are viewed as modifying the document term weights W_{rmDi}. It is regarding this approach that Fuhr stated "because there hardly will be enough relevance information available to estimate the probabilities....all attempts in this direction are doomed to fail" (!)(Fuhr and Buckley 1991, July). We know of no such attempts.

In the probabilistic models, the retrieval function $f_{i,j}$ utilizes relevance feedback data, so there is no separation between retrieval and learning, as there is in VSM. Because learning and retrieval are intertwined in the probabilistic models, the original Robertson et al. (Robertson, Maron et al. 1982) unification of the two methods of *learning*, was referred to as "A Unification of Two Competing Models for Document *Retrieval*". In section 4 we present a re-unification of the two competing approaches to learning for both the probabilistic and Vector Space models. In the case of the probabilistic models, it is appropriate to view this as a unification of two competing models for document retrieval.

In the next section, we present a shortcoming of all the above-described approaches to feedback learning. *Our primary aim is to improve and test methods of permanent learning for VSM and the binary probabilistic models, as these have received little attention in the past, and moreover, unlike transient feedback, they can improve a system's first-iteration retrieval results, even before eliciting user feedback.* This is especially important in light of experimental evidence that many retrieval results do not provide any relevant documents in the first iteration (Harman 1992. November 4-6), making transient feedback impossible, and frustrating users who may give up after such a failed initial system response.

3 The Credit Assignment Problem

Both VSM and the probabilistic models restrict themselves to learning about one of two objects in the network model, either weights W_{rmDi}, or weights W_{Qlcm}. In both VSM and probabilistic approaches, systems which aim to implement transient feedback, use the relevance feedback (D_i, Q_j, R) to estimate W_{Qlcm}, whereas in systems which aim for permanent learning, the user's feedback is used to estimate W_{rmDk}.

Reference to the network model shows that using feedback data to estimate either W_{Qlcm} exclusively or W_{rmDi} exclusively, represents a failure to include for estimation all the variables of the network. Feedback triples (D_i,Q_j,R_ij)

regard a single top-level document node and the single query node. This user feedback does not specify which individual network parameter is responsible for the positive or negative feedback. If the feedback data is to be optimally used, all these network parameters must be included in a single parameter estimation problem, so that for each feedback data point, an attempt can be made to assign an appropriate amount of credit to each parameter in the model.

Transient and permanent learning are both necessary because document terms are not always accurately indicative of the actual document meaning, and query terms are not always indicative of the user's true meaning. Moreover, recent work in user modeling has shown that users may modify their queries and query terms (Bates 1989), immediate search goals ((Tenopir, Nahl-Jakobovits et al. 1991), as opposed to (Hert 1996)), or even their overall information need (Katzer and Snyder 1990), during a single search session. Thus, the user's initial query terms as submitted to the retrieval function may not accurately reflect his current information need. Regarding *document* terms, not every term in a document is a proper representative of the document's meaning, due in part to the complexities of language such as polysemy, synonymy, phrases, and context.

When a user supplies feedback of the form (D_i, Q_j, $R_{i,j}$), the retrieval system knows with certainty only that document D_i is relevant to this user's information need. But what may the system *learn* from this information? Should it assume that the document terms are a perfect representation of its actual contents, and learn that the user's need is similar to the contents of D_i? In this case, the system performs transient feedback which corrects the weights W_{Qlcm}. Or should it instead assume that query terms are a perfect representation of the user's information need, and learn that the document is actually about that topic? In this case, the system performs permanent learning which corrects the weights W_{cmDi}. These choices create a credit assignment problem. In order to optimally use the available relevance data, a learning algorithm must include all the unknown parameters, and then must assign to each an appropriate amount of credit for the observed data.

4.0 Optimal Solutions

We are given feedback data of the form (D_i, Q_j, $R_{i,j}$). Let $f(D_i,Q_j)$ represent the function which gives a relevance score for a document D_i with respect to a query Q_j.

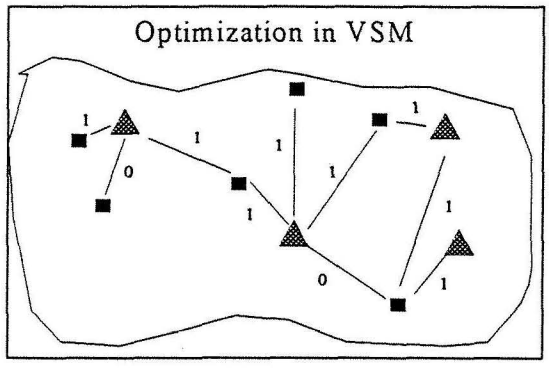
We can formulate a straightforward (if ideal) optimization problem to reduce error. Assuming system output is normalized to [0,1], and relevance assessments are binary, the error we wish to reduce is $\Sigma\Sigma(R_{i,j} - f(D_i,Q_j))^2$. Identically, if relevance assessments are 1 or -1, then we wish to maximize $\Sigma\Sigma\ R_{i,j} * f(D_i,Q_j)$. In the following sections we apply this approach to the VSM and probabilistic retrieval models, in turn.

4.1 Optimization in VSM

For VSM, the ideal optimization formula takes the form: Find values for Di1..Din, Qj1..Qjn by minimizing $\Sigma\Sigma(R_{i,j} - D_i*Q_j)^2$ subject to the constraint $|D_i| = |Q_j| = 1$. In an n-dimensional space, an arbitrary additional n points also need to be fixed in position, to prevent rotations of the solution.

Figure 4 depicts this optimization problem for VSM. Every document-query pair about which feedback data is available is connected with a link labeled 1 or 0, for relevance or non-relevance. The aim is to maximize: the distance between all irrelevant pairs minus the distance between all relevant pairs.

Figure 4



It is important to note that this formulation is different from the traditional transient and permanent learning in two ways. First, a solution is found using *both* sorts of learning simultaneously. Second, the traditional approaches did not even find *optimal* positions for documents or queries individually. Rather, the queries and documents were moved, in the traditional approaches, according to pre-defined formula such as $Q_{new} = Q_{old} + \Sigma_{Di} - D_j$, as mentioned above. So our approach introduces the goal of optimization in general, as well as the goal of *simultaneous* optimization of both documents and queries.

The number of parameters to be estimated in this approach is prohibitive. Nevertheless, this formulation is useful, if only because it presents the ideal of simultaneously treating query and document parameters in the estimation process. We can severely limit the number of parameters to be estimated with any number of simplifying assumptions. For example, we may specify that all terms which are zero in the original document and query representation, remain fixed at zero. Just this assumption, which is analogous to assumptions commonly made in application of the probabilistic models (Fuhr and Buckley 1991, July), dramatically reduces the number of parameters.

4.2 Optimization in Probabilistic Approaches

The optimal solution for the probabilistic case is equally straightforward, but its relationship to the traditional approaches is more complex. The optimal solution -- assuming again system outputs normalized to [0,1] and binary relevance assessments -- is to minimize (for BIR) $\Sigma\Sigma(R_{i,j} - \Sigma D_{im}*W_{Qlm})$. There is a complication, however. The weights W_{Qlm} cannot be blindly estimated, because they are *derived* from the underlying contingency tables, which, in turn, are derived

from the joint data on relevance, document terms, and query terms.

Our approach, which is somewhat radical for the binary probabilistic models, is to estimate the values in Table 1 above -- i.e. the presence of terms in documents and queries -- to maximize the above expression. This is perfectly analogous to the VSM learning approach in which document and query term weights are learned. The probabilistic approaches, on the other hand, have traditionally not used feedback data to inform the model about the presence or strength of terms in individual documents and queries, but only to estimate the probability of relevance of one to the other. This radical departure has come about, because in the probabilistic models, transient and permanent learning amount to marginal views on joint data, and it makes no sense to search for optimal marginal values, except by estimating the underlying joint data. The meaning of estimating the term values in each query and document, is that we are estimating what terms would have apparently best represented the document or query, given the total available feedback data.

The ideal optimization problem in a probabilistic approach also has far too many parameters, but the problem may be constrained with any number of simplifying assumptions. First, as previously suggested for VSM, we could fix all terms which are initially zero. But the probabilistic models lend themselves to a kind of simplifying assumption which has no corollary in the VSM. Analogous to Robertson's Model 2, we could further severely constrain the problem space by requiring all $D_{im}=D_{jm}$ for all i,j. We would then be estimating the probability that an *arbitrary* document containing term m is properly indexed with that term. A similar constraint could hold instead for $Q_{im}=Q_{jm}$ for all i,j, analogous with Robertson's Model 1.

A drawback of the optimization formulations is the difficulty of incorporating incremental new evidence without re-evaluating the whole set of parameters. Related to this is the difficulty of specifying default parameters. Therefore, although the optimization formula may be feasible with strong simplifying assumptions, intelligent heuristics may be preferable.

5.1 Heuristics for Improving VSM Permanent Learning

The basic approach of our heuristics is that permanent learning can be improved by also accounting for transient feedback (and vice versa, but our emphasis here is on methods for improving permanent learning).

The first suggested heuristic is this: Before correcting a document by moving it towards query vectors (permanent learning), correct those query vectors first, using transient feedback ! This approach is literally a first-pass approach to the optimal, and can be extended backward any number of steps, so that each document is moved toward queries which are themselves first moved toward other documents which are themselves first moved....and so on.

This approach retains the notion of accounting for both transient and permanent learning, by applying both in series, but completely abandons the aim of optimization, and instead reverts back to predefined formula of the traditional sort.

A less efficient alternative which does retain the aim of optimization, is to perform an actual local optimization by finding the optimal positions -- using the formula in section 4.1 above -- for just the document in question and each of the queries for which we have feedback data R_{kj}. This amounts to drawing a sort of radius around document D_k to include all queries related to it, and to apply the optimization to just those queries and the one document. This radius can be drawn to any size, so that a document, its related queries, and their related documents, etc., are all included, for any number of degress of separation.

5.2 Heuristics for Improving Probabilistic Feedback Learning

Our heuristic for accounting for both marginal views, it this: Rather than merely counting the *number* of queries containing term m for which this document was relevant/irrelevant, we consider as well the *weights* of those query terms in their respective queries.

Here is a concrete example: Suppose we have the following contingency table for D_{1m}

Table 5: $Term_m$ in Document D_1

	Relevant	non-Relevant
Qjm=1	1	2
Qjm=0	1	1

The traditional weight for this term would be derived as follows:

Weight = $\log[p*(1-q)/q*(1-p)] = \log[(1/2*1/3) / (2/3*1/2)] = \log 1/2 \approx -0.3$

But suppose, instead of merely counting the number of queries, we account also for the weights of those query terms in their respective queries. Our modified contingency table might then look as follows, with each query term Q_{jm} being represented by its weight, computed in the traditional manner:

Table 6: $Term_m$ in Document D_1, Revised

	Relevant	non-Relevant
Qjm=1	-1.2	2.04 1.90
Qjm=0	0.5	0.5

These query term weights are then used to derive the document term weight in question, as follows (the query term weights are used as exponents, because they were originally derived with logs):

$P(Q_{jm}=1|R,D_1)=10^{-1.2}/(10^{-1.2}+10^{0.5}) \approx .02$

$P(Q_{jm}=1|notR, D_1) = (10^{2.04} + 10^{1.90}) / (10^{2.04} + 10^{1.90} + 10^{0.5}) \approx .97$

The weight of term m in document one is therefore $\approx \log (.02*.03)/(.97*.98) \approx -3.2$

The reason for the dramatic change, in this example, of the weight of this document term, is that we have incorporated the fact that for the one query containing term m for which D_1 was

relevant, term m was *not* a good query term, while for the two queries containing term m for which D_1 was irrelevant, term m *was* a very good query term.

This first heuristic is analogous to the VSM approach of first modifying the query vectors, then modifying the document vector. Here, we account for the query term weights when computing the document term weight. In both the VSM and probabilistic cases, if one were more concerned with improved transient feedback, then the order of application of query and document estimates would be reversed in these heuristics.

The probabilistic analog to the second VSM heuristic, is to apply the optimization formula, except to estimate only the value of terms in the given document and all queries for which we have relevance data for that document.

5.3 Unification of Models 1 and 2

The optimal and heuristic approaches to permanent learning are unifications of Robertson et al's Models 1 and 2, in a way that differs from the unification Models 0 and 3 in (Robertson, Maron et al. 1982). Essentially, our unification of models 1 and 2 directly relates and improves the two. In the optimal solutions, the transient feedback approach of Model 2 and the permanent learning approach of Model 1 are simulatneously accounted for. In the heuristic solutions, both of these approaches are again accounted for, but in series, rather than in parallel, according to one heuristic, and in part, rather than in whole, according to the second heuristic of local minimization. Our approach, then, aims to *improve the estimates of models 1 and 2*. Robertson et al's Model 3, on the other hand, requires the *usual, separate* marginal estimates of models 1 and 2, and *uses* these to refine the estimates of relevance of a particular document to a particular query in Model 3. Among other differences, our approach is *applicable for the first round of a user query*, to improve retrieval results through permanent learning, before any feedback data is available for the query, while for Robertson et al.'s unification, Model 2 query feedback data must be available for Model 3 unification.

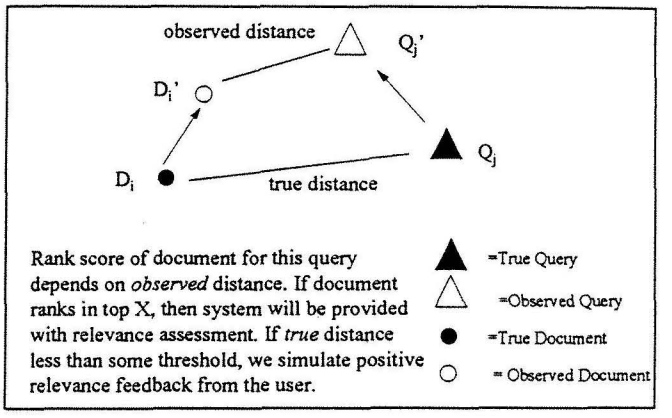
6 Simulation

To test these ideas in a controlled manner, in addition to the proposed field study, we have implemented a simulation study. A primary reason for implementing this simulation is that there does not exist any data set with sufficient relevance feedback data to support the sort of permanent learning addressed in this paper. The TIPSTER collection, for example, has an average of approximately one fifth of one data point per document in the TREC3 relevance assessments. The field study outlined below will provide sufficient feedback, but until that data is accumulated, it was believed that a simulation would be helpful. We have executed the simulation for VSM learning only.

The approach is as follows: A large number of document (D_i) and query vectors (Q_j) are generated, with each element being drawn randomly from a univariate normal distribution. These vectors are considered the "true" documents and queries. The score of each document with respect to each query is computed as the usual inner product D_i*Q_j. A document is considered relevant to a query if its score is above some threshold: $D_i*Q_j >$ *rel_cutoff*, that is, if the true distance between the document and query is below some threshold. Then, to simulate the imperfections in both document and query representations, we add a random vector of error to each document and query, producing "observed" documents D_i' and queries Q_j', according to traditional statistical theory of "observed=true+error". This error vector is also drawn from univariate normal, but is then multiplied by a factor *e_rate*, which controls the magnitude of the error.

Next, we model the process of user-supplied partial relevance feedback, by simulating that a user reviews and supplies feedback for the top x number of retrieved documents (not all of which are necessarily relevant). To simulate realistic relevance feedback, the documents are ranked for each query according to their VSM scores using the *observed* document and query vectors D_i' and Q_j', since the system uses only these *observed* vectors in calculating rank scores. The documents are therefore ranked according to their *observed* distance. The user is then simulated to review the top x-ranked documents based on observed distance, and to consider relevant any document (among those x documents) whose *true* distance is below some threshold, according to $D_i*Q_j >$ *rel_cutoff*. Negative relevance assessments were not used in this simulation. Figure 5 depicts this process.

figure 5



We are then in a position to test the various methods of permanent learning. First, we calculate the total initial distance of all observed documents from their true positions -- i.e. the total actual document error. As each method is applied to using feedback data for learning, we measure the extent to which the observed documents have been moved closer to their true locations (we have also measured traditional recall and precision measures, which roughly correspond, but considered that these may be misleading in this highly artificial simulation). As of this writing, we have tested only the traditional approach against the first VSM heuristic. The formula adopted for the traditional and new approaches to document modification is $Di_{new} = Di_{old} + \alpha(Ql - Di_{old})$ (Brauen 1971), while the formula adopted for

the query modification stage of our heuristic is $Q_{new} = Q_{old} + \sum D_i$, a simplified version of the usually better-performing $Q_{new} = Q_{old} + \sum D_i - D_j$ (see section 2.1).

Preliminary results are interesting and very encouraging. The benefits of our heuristic for permanent learning -- i.e. first correct the queries, then the documents -- depend on two factors: the size of the query error relative to the document error, and the amount of data available to correct each query relative to the amount of data available to correct each document. The following table depicts these initial findings of the relative performance gain of the proposed heuristic over traditional document modification:

Table 7: Performance Gains using Proposed Heuristic

	query e_rate = .4 doc e_rate = .2	query e_rate = .2 doc e_rate = .2
1000 Queries 1000 Documents	+ 175 %	+ 12 %
1000 Queries 500 Documents	+ 113 %	+ 4 %

This simulation contains many parameters: 1) number of queries, 2) number of documents, 3) query error rate, 4) document error rate, 5) number of dimensions per vector, 6) number of documents "reviewed" per query for feedback, 7) rel_cutoff, and 8) α from the traditional formula. The results shown in Table 7 show varying values for the first four parameters, given 100 dimensions and 10 documents reviewed per query, α=.2 (for the document modification formula) (Brauen 1971), and *rel_cutoff* = .25 (2.5 * the standard deviation of scores).

The benefits of our approach appear to depend on the relative size of the document and query errors, as well as on the absolute and relative numbers of documents (queries) available to correct each query (document). For example, in the first column of table 7, lesser (although still very significant) improvement is shown where the number of queries is twice the number of documents (1000 queries versus 500 documents). In the simulation run corresponding to that lower-left cell in table 7, there were available 2,842 data points of positive relevance feedback. Therefore, 2842/1000 $\approx$ 3 data points were available, on average, to correct a given query, while 2842/500 $\approx$ 6 data points were available to correct each document. In this instance, there remains great benefit to correcting queries before documents, but the benefit is somewhat diminished, because the structure of available data makes query-fixing less necessary than otherwise, and also relatively less reliable than the document-fixing.

The second axis of table 7 regards the error rates of documents and queries. We anticipate that in reality, there may be much more error in the queries, because of the very small number of terms per query, relative to the number of terms per full-text-indexed document. With this and further uses of the simulation, we gain an

understanding of the factors which affect the performance of learning in general, and of our improvements in particular. A field study remains vital to measure the levels of these factors in a real setting -- e.g. what is the actual level of "error" in queries versus documents.

7 Hypotheses and Ongoing Field Study

A field study is under way to test the following hypotheses which result from this work. We include this summary of hypotheses to emphasize the expected contributions of the theory previously covered.

The first hypothesis of this study is that permanent learning is possible and effective according to the traditional VSM and probabilistic models. VSM permanent learning has been little studied, and probabilistic permanent learning of the BII model has not been studied at all. In the case of BII, Fuhr has warned against even attempting this approach due to the scarcity of available feedback data. We believe this admonition was premature, since we can indeed now collect from users sufficient data to implement and test the VSM and probabilistic versions of permanent learning. We state a null hypothesis (H1) that permanent learning is feasible and effective (performance measures discussed below). A second hypothesis regards the relative effectiveness of VSM and probabilistic permanent learning. Due to lack of space, we have not discussed the factors which might favor one system over the other, but we will state a null hypothesis (H2) that both are equally effective. The third and fourth hypotheses of this study, about which we have gone into great detail, is that both transient and permanent learning can be improved by applying both together. We will investigate both optimal and heuristic approaches. We hypothesize that the heuristic approaches will outperform the traditional (H3), and that the optimal approaches will outperform the heuristics (H4). Finally, an additional benefit of the field study is that we will have real data on the absolute baseline performance of a full-text retrieval system in the field. Such data is scarce. We state null hypothesis H5: Users assess that full-text retrieval system works flawlessly.

The following table depicts all the system types which need to be tested, to address all the hypotheses. The simulation reported above regards the VSM aspect of H3.

Type of Permanent Learning

	None	Traditional	Heuristic-Improved	Optimal
VSM	H5	H1, H2, H3, H4	H3, H4	H4
Probab-ilistic	H5	H1, H2, H3, H4	H3, H4	H4

The field study setting is the monitored use of the SEC EDGAR database, http:\\edgar.stern.nyu.edu. We will make available through full-text indexing all 8K's (press releases) 10K's (annual reports), and 14A's (proxy statements). Simple feedback data will be collected from users. This data will be used by each approach to learning. A small sample of queries will not be used for learning,

but will instead be used as "test" queries. Retrieval performance will be judged by traditional recall and precision estimates for the sample of test queries. Results from this study will have the added benefits of external validity, as well as supplying valuable information regarding how the SEC documents are used. We will also then have a data set with sufficient user-supplied relevance feedback to allow application of -- and new research into -- learning methods which were previously considered infeasible.

Summary

The era of large full-text databases is here, and the value of good text retrieval will certainly increase in the coming years. Fortunately, the era of Web logs is also here, so that we may for the first time capture an extraordinary amount of information about our users' queries and their evaluations of the retrieved documents. This user feedback also allows us to implement and test theories of permanent, continuous learning, which were previously considered impractical due to the scarcity of feedback data. In addition. this study introduces the notion of optimization to the IR learning problem. This formulation of IR learning as an optimization problem should allow application to this problem of well-established mathematical techniques, which may be of great value to improving text retrieval in the coming years.

References

Bartell, B., G. W. Cottrell, et al. (1992). Latent Semantic Indexing is an Optimal Special Case of Multidimensional Scaling. 15th Int'l SIGIR, Denmark.

Bates, M. J. (1989). "The Design of Browsing and Berrypicking Techniques for the Online Search Interface." *Online Review* **13**(5): 407-424.

Belew, R. K. (1989). "Adaptive Information Retrieval: Using a connectionist representation to retrieve and learn about documents." : 11-20.

Bookstien, A. and D. R. Swanson (1974, September/October). "Probabilistic Models for Automatic Indexing." *Journal of the American Society for Information Science* **25**(5): 312-376.

Brauen (1971). Document Vector Modification. The SMART Retrieval System: Experiments in Automatic Document Processing. G. Salton. Engelwood Cliffs, N.J., Prentice Hall: 456-484.

Cavnar, W. B. (1993). Using An N-Gram-Based Document Representation With A Vector Processing Retrieval Model. TREC3, Gaithersburg, MD.

Cooper, W. S., A. Chen, et al. (1995). Experiments in the Probabilistic Retrieval of Full Text Documents. TREC3, Gaithersburg, MD., National Institute of Standards and Technology.

Cooper, W. S., F. C. Grey, et al. (1992). Probabilistic Retrieval Based on Staged Logistic Regression. 15th Int'l SIGIR, Denmark.

Fuhr, N. (1989). "Models for Retrieval with Probabilistic Indexing." *Information Processing and Management* **25**(1): 55-72.

Fuhr, N. and C. Buckley (1991, July). "A Probabilistic Learning Approach for Document Indexing." *ACM Transactions on Information Systems* 9(3): 223-248.

Harman, D. K. (1992). Relevance Feedback Revisited. 15th Int'l SIGIR, Denmark.

Harman, D. K. (1992. November 4-6). The First Text Retrieval Conference (TREC-1), Gaithersburg, Maryland, National Institute of Standards and Technology

Harper, D. J. and C. J. V. Rijsbergen (1978, September). "An Evaluation of Feedback In Document Retrieval Using Co-Occurrence." *Journal of Documentation* **Vol. 34**(Number 3): 189-216.

Harter, S. P. (1975). "A Probabilistic Approach to Automatic Keyword Indexing Part 1: On the Distribution of Specialty Words in a Technical Literature." *Journal of the American Society for Information Science* **September-October**: 197-206.

Harter, S. P. (1975). "A Probabilistic Approach to Automatic Keyword Indexing Part Two: An Algorithm for Probabilistic Indexing." *Journal of the American Society for Information Science* **September-October**: 280-289.

Hert, C. A. (1996). "User Goals on an Online Public Access Catalog." *Journal of the American Society for Information Science* 47(7): 504-518.

Katzer, J. and H. W. Snyder (1990). Toward a More Realistic Assessment of Information Retrieval Performance. Annual Meeting of the American Society for Information Science, Learned Information.

Kwok, K. L. (1989). "A Neural Network for Probabilistic Information Retrieval." : 21-30.

Lewis, D. D. (1992, February). Representation and Learning in Information Retrieval. Computer and Information Science. Massachusetts, University of Massachusetts: 280.

Maron, M. E. and J. L. Kuhns (1960). "On Relevance, Probabilistic Indexing and Information Retrieval." *Journal of the ACM* 7: 216-244.

Masand, B., G. Linoff, et al. (1992). Classifying News Stories using Memory Based Reasoning. 15th Int'l SIGIR, Denmark.

Rijsbergen, C. J. v. (1975). Information Retrieval. Boston, Buttersworth.

Robertson, S. E. (1977, December). "The Probability Ranking Principle in IR." *Journal of Documentation* **33**(4): 294-304.

Robertson, S. E. and K. S. Jones (1976, May-June). "Relevance Weighting of Search Terms." *Journal of the American Society for Information Science* **27**(3): 129-146.

Robertson, S. E., M. E. Maron, William S. Cooper (1982). "Probability of Relevance: A Unification of Two Competing Models For Document Retrieval." *Information Technology and Libraries*: 1-21.

Salton, G. (1989). Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer, Addison-Wesley.

Salton, G., Chris Buckley (1990). "Improving Retrieval Performance by Relevance Feedback." *Journal of the American Society for Information Science* **41**(4): 288-297.

Salton, G. and C. Buckley (1988). "Term-Weighting Approaches in Automatic Text Retrieval." *Information Processing and Management* **24**(5): 513-523.

Salton, G., C. S. Yang, et al. (1975, January-February). "A Theory of Term Importance in Automatic Text Analysis." *Journal of the American Society for Information Science* **26**(1): 33-43.

Tenopir, C., D. Nahl-Jakobovits, et al. (1991). "Strategies and Assessments Online: Novices' Experience." *Library and Information Science Research* **13**: 237-266.

Turtle, H., Bruce Croft (1991). "Evaluation of an Inference Network-Based Retrieval Model." *ACM Transactions on Information Systems* **9**(3): 187-222.

Yu, C. T., W. S. Luk, et al. (1979). "On Models of Information Retrieval Processes." *Information Systems* **4**: 205-218.