A survey on the use of relevance feedback for information access systems

Ian Ruthven
Department of Computer and Information Sciences
University of Strathclyde, Glasgow, G1 1XH.
Ian.Ruthven@cis.strath.ac.uk

Mounia Lalmas
Department of Computer Science
Queen Mary, University of London, London, E1 4NS.
mounia@dcs.qmul.ac.uk

Abstract

Users of online search engines often find it difficult to express their need for information in the form of a query. However, if the user can identify examples of the kind of documents they require then they can employ a technique known as relevance feedback. Relevance feedback covers a range of techniques intended to improve a user's query and facilitate retrieval of information relevant to a user's information need. In this paper we survey relevance feedback techniques. We study both automatic techniques, in which the system modifies the user's query, and interactive techniques, in which the user has control over query modification. We also consider specific interfaces to relevance feedback systems and characteristics of searchers that can affect the use and success of relevance feedback systems.

1 Introduction

Information retrieval (IR) systems allow users to access large amounts of electronically stored information objects [VR79, BYRN99, Bel00]. A user submitting a request to an IR system will receive, in return, a number of objects relating to her request. These objects may include images, pieces of text, web pages, segments of video or speech samples.

A number of features distinguish IR systems from other information access tools. For example, an IR system does not extract information from the objects that it accesses. Neither, typically, does it process information contained within these objects. This separates IR systems from knowledge-based systems such as expert systems, conceptual graphs or semantic networks. These knowledge-based tools depend heavily on a pre-defined representation of a domain, such as medicine or law. This domain knowledge can be used to manipulate, infer or categorise information for a user. Instead, IR systems are used to direct the user to objects that may help satisfy a need for information.

The data accessed by IR systems is usually unstructured, or at best semi-structured. The requests submitted to IR systems are generally also unstructured. Whereas a database system will be used to answer requests such as "*How many female members of parliament are there in the British Parliament?"* or *"Which British MPs are women?",* IR systems will be used to answer requests such as *"What are the main causes of the poor representation of women in UK politics?"* or *"In what ways are the British political parties attempting to increase the number of female MPs".* IR systems are intended to deal with requests that do not necessarily specify a unique, objective answer.

The process of IR is, therefore, an inherently *uncertain* one. Searchers may not have a well-developed idea of what information they are searching for, they may not be able to express their conceptual idea of what information they want into a suitable query and they may not have a good idea of what information is available for retrieval. Early in the field, researchers recognised that, although users had difficulty expressing exactly the information that they required, they could recognise useful information when they saw it. That is, although searchers may not be able to convert their need for information into a request, once the system had presented the user with an initial set of documents the user could indicate those documents that did contain useful information.

This lead to the notion of *relevance feedback* (RF): users marking documents as *relevant* to their needs and presenting this information to the IR system. The system can then use this information quantitatively - retrieving more documents like the relevant documents - and qualitatively - retrieving documents similar to the relevant ones before other documents. The process of RF is usually presented as a cycle of activity: an IR system presents a user with a set of retrieved documents, the user indicates those that are relevant and the system uses this information to produce a modified version of the query. The modified query is then used to retrieve a new set of documents for presentation to the user. This process is known as an *iteration* of RF.

The mechanism by which an IR system uses the relevance information given by the user is the main focus of this paper. The paper covers several aspects of RF: the representations used in RF, how these representations lead to deciding how to modify a query and the role of interaction in RF. Section 2 presents a brief discussion of the retrieval process as a whole and outlines how RF has been incorporated into the major retrieval models. In section 3 we discuss extensions and modifications to the traditional models of RF.

Historically, most RF approaches have been based on *automatic* techniques for modifying queries. In section 4 we summarise these approaches. More recently, a number of researchers have examined the role of the user in RF and have presented techniques designed to increase the interaction between the user and system in RF. These *interactive* techniques are the main topic of section 5. In section 6 we describe interfaces specifically designed to facilitate RF, in section 7 we outline some of the important aspects the user that are important to RF, and we conclude this overview in section 8.

2 The information retrieval process

The IR process is composed of four main technical stages. The first stage, *indexing* the document collection, during which the documents are prepared for use by an IR system, is discussed in section 2.1. Document *retrieval*, the process of selecting which documents to display to the user, is described in section 2.2. The *presentation* of retrieved documents and the *evaluation* of the retrieval results are discussed briefly in sections 2.3 and 2.4 respectively. In the section on retrieval we shall outline the basic approaches to RF in the major retrieval models. In section 2.5 we shall summarise the difference between these main approaches to RF.

2.1 Indexing

For small collections of documents it may be possible for an IR system to assess each document in turn, deciding whether or not it is likely to be relevant to a user's query. However, for larger collections, especially in interactive systems, this becomes impractical. Hence it is usually necessary to prepare the raw document collection into an easily accessible representation; one that can target those documents that are most likely to be relevant, for example those documents that contain at least one word that appears in the user's query.

This transformation from a document text to a *representation* of a text is known as *indexing* the documents. There are a variety of indexing techniques but the majority rely on selecting good document descriptors, such as keywords, or *terms*, to represent the information content of documents. A 'good' descriptor for IR is a term that helps *describe* the information content of the document but is also one that helps differentiate the document from other documents in the collection. A 'good' descriptor, then, has a certain *discriminatory* power[1]. This power of a term in discriminating documents can be used to differentiate between relevant and non-relevant documents, as will be discussed in the section on retrieval.

Figure 1 outlines the basic steps in transforming a document into an indexed form. The first stage is to convert the document text (**Document text**, Figure 1a) into a stream of terms, typically converting all the terms into lower case and removing punctuation characters (**Tokenisation**, Figure 1**b**).

[1]See [VR79], Chapter 2, for a more detailed explanation of the trade-off between the descriptive and discriminatory power of terms.

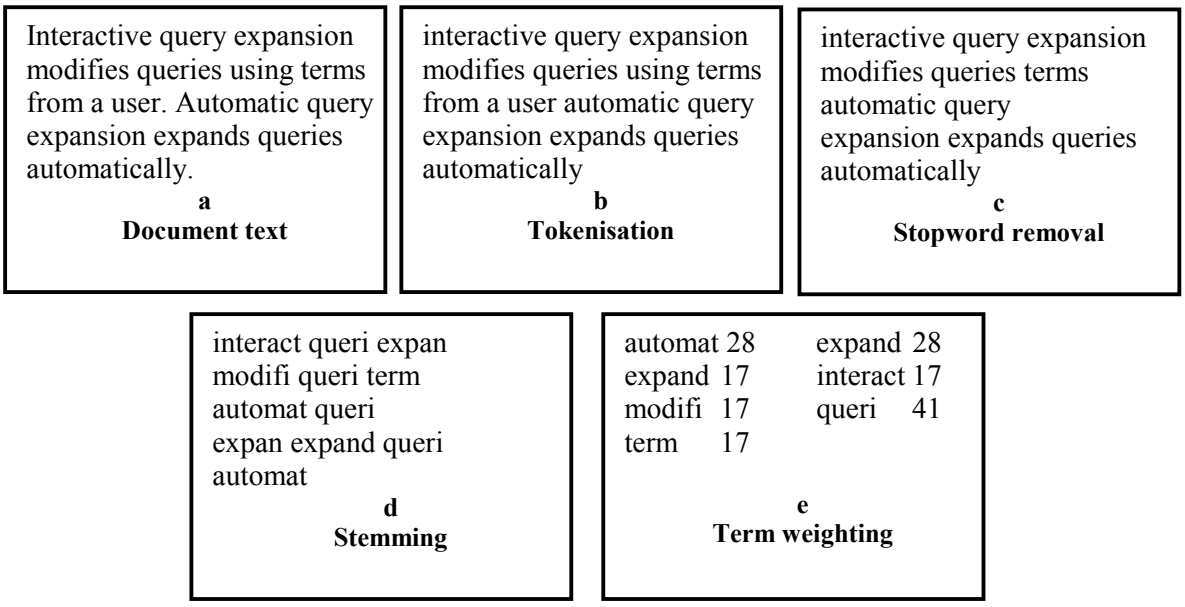

Figure 1: Indexing a document

Once the document text has been tokenised it is necessary to decide which terms should be used to represent the documents. That is, we need to decide which descriptors are useful for the joint role of describing the document's content and discriminating the document from the other documents in the collection. Very high frequency terms, ones that appear in a high proportion of the documents in the collection, tend not to be effective either in discriminating between documents or in representing documents.

There are two main reasons for this. The first is that, for the majority of realistic user queries, the number of documents that are *relevant* to a query is likely to be a small proportion of the collection. A term that will be effective in separating the relevant documents from the non-relevant documents, then, is likely to be a term that appears in a small number of documents. Therefore high frequency terms are likely to be poor at discriminating. The second reason is related to the notion of *information content*. Terms that can appear in many contexts, such as prepositions, are not generally regarded as *content*-bearing words; they do not define a topic or sub-topic of a document. The more documents in which a term appears (the more contexts in which it is used) then the less likely it is to be a content-bearing term. Consequently it is less likely that the term is one of those terms that contribute to the user's relevance assessment. Hence, terms that appear in many documents are less likely to be the ones used by a searcher to discriminate between relevant and non-relevant documents.

A common indexing stage is, then, to remove all terms which appear commonly in the document collection, and which will not aid retrieval of relevant material, (**Stopword removal**, Figure 1**c**). The list of terms to be removed is known as a *stop-list*; these can either be generic lists, ones that can be applied to most collections, e.g. [VR79], or lists that are specifically created for an individual collection. A term does not have to appear in the majority of documents to be considered a stop term. For example, in [CRS+95] the removal of all terms that appeared in more than 5% of documents did not significantly degrade retrieval performance in a standard IR system.

Terms may appear as linguistic variants of the same word, e.g. in the example in Figure 1, the terms *queries* and *query* are the plural and singular of the same object and the terms *expansion* and *expand* refer fundamentally to the same activity. As most IR systems rely on functions that *match* terms (see section 2.2) to retrieve documents, this variation in word use could cause problems for the user. For example, if a user enters a query '*hill walks*' then an IR system will retrieve all documents that contain the term '*walks*' but not documents containing '*hill walking*', '*hill walk*' or '*hill walker*', any of which may contain relevant information. To avoid the user having to instantiate every possible variation of each

query term, many indexing systems reduce terms to their root variant, a process known as *stemming* [Por80] (**Stemming**, Figure 1**d**)[2].

The result of the indexing process, so far, is a list of low to medium frequency terms that represent the information content of the document and help discriminate the document from other documents. This information can be included in a file containing the information on all the document collection, known as an *inverted file*, Figure 2. In this file each line consists of information on one of the terms in the collection; in this example we have the term (*automat*), followed by a series of document identifiers.

automat	1	2	3 ....
expan	1	4	6 ....
expansion	1	17	46....
...			

Figure 2: Inverted file with no term weights

The final stage in most IR indexing applications is to weight each term according to its importance, either in the collection, in the individual documents or some combination of both, (**Term Weighting**, Figure 1**e**). Two common weighting measures are inverse document frequency (*idf*) [SJ72] and term frequency (*tf*) [Har92a]. *idf* (or as it is sometimes referred to, inverse collection frequency) weights a term according to the inverse of its frequency in the document collection: the more documents in which the term appears, the lower *idf* value it receives, Equation 1. The *idf* weighting function, then, assigns high weights to terms that have a high discriminatory power in the document collection.

$$idf(t) = \ln\frac{N}{n}$$

Equation 1: Inverse document frequency
where N = number of documents in the collection
n = number of documents containing the term t

Term frequency, or *tf*, measures (see [Har92a] for an overview) assign larger weights to terms that appear more frequently within an individual document. Unlike the *idf* value, the *tf* value of a term is dependent on the document in which it appears, Equation 2. The *tf* weighting function assigns high weights to terms that appear more frequently within a document.

$$tf_d(t) = \frac{\ln(occs_t)}{\ln(length_d)}$$

Equation 2: Term frequency
where $length_d$ = the number of terms in document d
$occs_t$ = number of occurrences of term t in document d

Term weighting information can be also be included in the inverted file; in Figure 3 we have the term (*automat*), its *idf* value (*36*), followed by a series of tuples of the form <document identifier, *tf* value>

automat 36	<1, 28> <2, 14> <3, 28>	
expan	14	<1, 28> <4, 15> <6, 29> ....
expansion	11	<1, 17>...
...		

Figure 3: Inverted file with *idf* and *tf* weights

Some kind of inverted file will form the main data structure of most IR systems and its use means that the IR system can easily detect which documents contain which query terms. Stopword removal and stemming reduce the size of the inverted file and increase the efficiency of the system.

[2]We shall continue to refer to stemmed terms as terms for ease of description.

Although indexing makes it possible to access information from very large document collections, the conversion from a document *text* to a list of weighted keywords does result in a loss of information. Writing a document is an intentional process; a document is intended to convey a message. The translation to a list of keywords retains the essential building blocks of the message, the terms themselves, but the message(s) that the author intended cannot be accessed by the retrieval mechanism. The effect of this loss of information may be ameliorated or deteriorated by the use of controlled vocabularies - pre-defined sets of indexing terms, [Ing92, Chap 3]. However, the fact remains that when we talk of representing the information content of documents we are only representing the *components* of the message, not the message itself.

The reduction of the document text into a series of keywords also transforms the task of an IR system from retrieving *information* to retrieving *objects* that contain information. Some authors argue that objects such as documents cannot be held to contain information as such, rather information is a change in a cognitive, or internal, state brought about by exposure to the contents of these objects. The following early quote by Maron, [Mar64], illustrates this concern,

> "..information is not a *stuff* contained in books as marbles might be contained in a bag - even though we sometimes speak of it in that way. It is, rather a *relationship*. The impact of a given message on an individual is *relative* to what he already knows, and of course, the same message could convey different amounts of information to different receivers, depending on each one's internal model or map."

The degradation of the document text, necessary for computation, and the subjectivity of relevance results in a layer of indirection between the user and the documents. The goal of the IR system is to bridge this gap between the user and potentially relevant material. Indexing techniques identify and highlight potentially good indicators of relevant material, and retrieval techniques use these indicators of relevance to select which documents to present to the user. *How* individual retrieval systems use these indicators to retrieve documents is the topic of the next section.

2.2 Retrieval and feedback

Retrieval is the process of *matching* a representation of an information need, usually a user-supplied *query*, to an indexed document representation. Queries will be indexed in the same way as a document and compared with a document index to determine if a document is likely to be relevant to a query. How the indexed query is compared with the indexed document differentiates the major retrieval models. In this section we shall briefly outline the four main models of retrieval: *Boolean*, *vector-space*, *probabilistic*, and *logical*, and describe the basic approaches to RF in each of the models.

2.2.1 Boolean model

The first operational IR retrieval model was the Boolean model, based on Boolean logic. In this model queries are keywords combined, by the user, with the conjunctive (AND), disjunctive (OR) or negation (NOT) operators. This is an *exact-match* model: the system only retrieves those documents that exactly match the user's query formula. For example, for the query *'information* AND *retrieval* AND *system'* the system will return all documents that contain the three words *'information*', '*retrieval'* and '*system'*, whereas the query *'information* OR (*retrieval* AND *system*)' will return those documents that contain the word *'information'* and those documents that contain both *'retrieval'* and *'system'*.

The Boolean model has been used in a large number of on-line public access catalogue (OPAC) systems but has been shown to demonstrate a number of difficulties. Firstly, traditional Boolean systems do not use term weights and consequently return the complete set of documents that match the query as an unordered set. This means the users may have to add or remove terms, or generate more complex query expressions to reduce the set of retrieved documents to a manageable size. Willie and Bruza, [WB95], argue that the problems with interacting with Boolean systems are not only a matter of the formal query language but a *conceptual* problem: the Boolean model does not lend itself to supporting how users think about searching and their individual search techniques. A further problem with Boolean systems is that the order in which operators are applied may not be consistent across systems, resulting in the fact that different systems may retrieve different documents for the same query, [Borg96]. Nevertheless Boolean systems do remain popular with users, perhaps because of the explicit control that is offered by these systems to the user. Web search engines often allow Boolean-style querying performed on an underlying best-match model (see section 2.2.2).

Harman [Har92c] suggests two possible methods for implementing RF on Boolean systems. The first is to present the user with a list of possible new query terms. These can be chosen, for example, by the term distribution in the relevant documents. This means selecting those terms that appear more often in the relevant than non-relevant documents and which would be useful to include in a new query. The second approach is for the system to automatically modify Boolean queries. An example of the latter type of query modification can be found in the system proposed by Khoo and Poo, [KP94], which is intended to automatically modify both the terms and the Boolean connectives of queries based on the documents marked relevant by a user.

An alternative to exact-match systems, such as the Boolean model, are *best-match* systems. These systems use term weights, such as *tf* and *idf*, to *rank* documents in decreasing order of matching score or estimation of relevance. The two most common best-match models are the *vector-space model*, which orders documents in decreasing *similarity* of query and document, [Sal71], and the *probabilistic model,* [RSJ76], which orders documents based on an estimate of the *probability of relevance* of a document to a query. In section 2.2.2 we discuss the vector space model, in section 2.2.3 we discuss the probabilistic model.

2.2.2 Vector space model

In the vector-space model, a document is represented by a vector of n weights, where n is the number of unique terms in the document collection. Figure 4 shows an example vector where x_i is the weight[3] of the ith term in document x if x contains the term, and 0 if the term is not present in x.

$$x = (x_1, x_2, ..., x_n)$$

Figure 4: Document vector

Queries are also represented as a vector of length n, and the similarity of the document vectors to a query vector gives a retrieval score to each document, allowing comparison and ranking of documents. A range of similarity measures exists to calculate this similarity, e.g. DICE, inner product, cosine correlation, [VR79, Chap 3]. Equation 3 shows the cosine correlation, one of the more common vector-space matching functions.

$$\cos(doc_i, query_j) = \frac{\sum_{k=1}^{n}(term_{ik} \cdot qterm_{jk})}{\sqrt{\sum_{k=1}^{n}(term_{ik})^2 \cdot \sum_{k=1}^{n}(qterm_{jk})^2}}$$

Equation 3: Cosine correlation between document doc_i and $query_j$

Unlike the Boolean model, which retrieves documents according to the query terms and query connectives, in the best-match models all documents that contain at least one query term will receive a non-zero score; the highest score going to documents that contain all the query terms. Documents that contain only some of the query terms will be ranked according to the sum of the weights of the query terms they contain. The documents that contain more query terms or contain query terms with a higher discriminatory power (term weight) will be retrieved above those that contain fewer query terms or query terms with lower weights. Similarity is then a function of term overlap between query and document, and the weights assigned to the terms.

Rocchio [Roc71] is generally credited with the first formalisation of a RF technique, developed on the vector space model. In [Roc71] he defines the problem of retrieval as that of defining an optimal query; one that maximises the difference between the average vector of the relevant documents and the average vector of the non-relevant documents. As discussed in section 1, it may not always be possible for a user to submit such an optimal query, so RF is required to bring the query vector closer to the mean of the relevant documents, and further from the mean of the non-relevant documents. This is

[3]Some implementations of the vector space model use 1 if a term occurs in a document, 0 if it does not occur. Most implementations will use some form of *tf***idf* weighting and some form of length normalisation will usually be performed to avoid retrieval bias towards long documents.

accomplished by the addition of query terms and by the reweighting of query terms to reflect their utility in discriminating relevant from non-relevant documents.

Rocchio's original formula for defining a new query vector in the vector space model, is as follows, Equation 4

$$Q_1 = Q_0 + 1/n_1 \sum_{i=1}^{n_1} R_i - 1/n_2 \sum_{i=1}^{n_2} S_i$$

Equation 4: Rocchio's original formula for modifying a query based on relevance information

where Q_0 = initial query vector, Q_1 = new query vector, n_1 = number of relevant documents, n_2 = number of non-relevant documents, R_i = vector for the ith relevant document, S_i = vector for the ith non-relevant document

The new query vector is the original query vector plus the terms that best differentiate the relevant documents from the non-relevant documents. A modified query contains new terms (from the relevant documents) and has new weights attached to the query terms. If the weight of a query term drops to zero or below, it is removed from the query.

This formula is capable of being constrained further, e.g. by weighting the original query vector so that the original query terms contribute more to the modified query than the new query terms or by varying the amount of feedback considered. A variation of this formula was tested experimentally with positive results on the SMART retrieval system [Roc71]. The small size of the document collection used in Rocchio's experiments meant that certain modifications had to be made to the formula. For example, although Rocchio tried to keep the size of the relevant and non-relevant feedback sets identical, this was not always possible. In addition a term was only considered if it was one of the original query terms or if it appeared in more relevant than non-relevant documents *and* in more than half the relevant documents. These modifications highlight the recurring difficulty of aligning theory with experimental practice.

Ide [Ide71] extended the SMART relevance feedback experiments, examining different aspects of RF, such as only using relevant documents for feedback, varying the number of documents used for RF, and using non-relevant documents. She found that using only relevant documents for feedback or varying the number of documents used at each iteration of feedback gave inconclusive or poor results.

Her third strategy was a variation of Rocchio's original formula, using only the first non-relevant document found, s_i. The formula used by Ide is shown in Equation 5. This was compared against Rocchio's original formula. Although this technique, the *Ide-dec-hi* formula, did not improve results greatly it was more consistent; improving the performance of more queries.

$$Q_1 = Q_0 + \sum_{i}^{n_r} r_i - s_i$$

Equation 5: Ide-dec-hi formula for modifying a query based on relevance information

where Q_0 = initial query vector, Q_1 = new query vector, n_r = number of relevant documents, r_i = vector for the ith relevant document, s_i = vector for the first non-relevant document

A common modification to the vector space RF formulae, e.g. [IdS71], is to weight the relative contribution of the original query, relevant and non-relevant documents to the RF process. In Equation 6, the α, β and γ values specify the degree of effect of each component on RF.

$$Q_1 = \alpha.Q_0 + \beta/n_1 \sum_{i=1}^{n_1} R_i - \gamma/n_2 \sum_{i=1}^{n_2} S_i$$

Equation 6: Rocchio modified relevance feedback formula

2.2.3 Probabilistic model

In the probabilistic model, suggested by Maron and Kuhns [MK60], and developed by amongst others, Robertson and Sparck Jones [RSJ76], and Van Rijsbergen [VR79], documents and queries are also viewed as vectors but the vector space similarity measure is replaced by a probabilistic matching function. The probabilistic model is based on estimating the *probability* that a document will be relevant to a user, given a particular query. The higher this estimated probability, the more likely the document is to be relevant to the user[4]. This is instantiated in the *probabilistic ranking principle*, [Rob77].

> "If a reference retrieval system's response to each request is a ranking of the documents in the collection in order of decreasing probability of relevance to the user who submitted the request, where the probabilities are estimated as accurately as possible on the basis of whatever data have been made available to the system for this purpose, the overall effectiveness of the system to its user will be the best that is obtainable on the basis of those data."

The estimated probability of relevance can be expressed as $P_q(rel \mid x)$, the probability of relevance given a document x and a query q. This probability can be used to decide whether or not to retrieve a document: if $P_q(rel \mid x) = 0$ then the probability of relevance given x is 0, and x should not be retrieved[5].

This can be refined by also considering the probability of non-relevance given x and q, $P_q(\overline{rel} \mid x)$. If $P_q(rel \mid x) > P_q(\overline{rel} \mid x)$ then it can be asserted that the probability of relevance is greater than the probability of non-relevance and hence x should be retrieved[6]. Thresholds may also be used, i.e. the difference between the probability of relevance and the probability of non-relevance must be greater than some threshold value before x is retrieved, $((P_q(rel \mid x) - P_q(\overline{rel} \mid x)) > threshold)$. In this case *threshold* is a value set by the user or system, in order to further restrict the retrieval function.

Having decided which documents to retrieve, the odds of relevance to non-relevance, Equation 7, can be used as a document *ranking* function: the higher the ratio of the probability of relevance to non-relevance, given x, then the more likely document x is to be relevant to a user.

$$\frac{P_q(rel \mid x)}{P_q(\overline{rel} \mid x)}$$

Equation 7: Odds of relevance to non-relevance for document x and query q

Bayes, [Bay63], theorem can be used to calculate $P_q(rel \mid x)$ and $P_q(\overline{rel} \mid x)$. Equation 8 demonstrates this for the relevance case.

$$P_q(rel \mid x) = \frac{P_q(x \mid rel) P_q(rel)}{P(x)}$$

Equation 8: Calculation of $P_q(rel \mid x)$ through Bayesian inversion

where $P_q(rel)$ is the prior probability that *any* document in the collection is relevant to q
$P_q(x \mid rel)$ is the probability of observing document x given relevance information
$P(x)$ is the probability of observing document x irrespective of relevance

---

[4]The probabilistic model measures the *probability* of relevance, i.e. the probability that a document will be relevant, not the *degree* of relevance as is sometimes suggested. A good discussion of the difference between these two notions is found in [RB78].

[5]In an operational system $P_q(rel \mid x)$ will generally only equal 0 if x does not contain any query terms. This rule then decides only to retrieve those documents that contain at least one query term.

[6]In the case where the two probabilities are equal, it is arbitrarily decided that x is non-relevant [VR79].

After Bayesian inversion and deletion of $P(x)$ (which is identical for both the relevance and non-relevance case), the odds function from Equation 7 turns into Equation 9**a**.

The probability of relevance, $P_q(rel)$, and the probability of non-relevance, $P_q(\overline{rel})$, are identical for all *x*'s, That is when we use the odds in Equation 7 to rank documents, the ranking is dependent on the values of the probabilities $P_q(x \mid rel)$ and $P_q(x \mid \overline{rel})$, not on the values $P_q(rel)$ and $P_q(\overline{rel})$. We can therefore eliminate these elements and arrive at the odds in Equation 9**b**. This is then the odds of observing *x* given relevance or non-relevance.

$$\frac{P_q(x \mid rel)P_q(rel)}{P_q(x \mid \overline{rel})P_q(\overline{rel})} \qquad \frac{P_q(x \mid rel)}{P_q(x \mid \overline{rel})}$$

a **b**

Equation 9: Odds of relevance, or non-relevance, having observed document *x*

The odds in Equation 9 refer to the probability of relevance, and non-relevance, after viewing the actual document text rather than the vector representation of the document. That is, it measures the odds of relevance to non-relevance based on the content of the document and is independent of the document representation. This means that the model can be used for many different types of document indexing but it also means that Equation 9 must be ultimately be expressed as a retrieval function based on the specific document indexing technique used to represent the documents.

There are many probabilistic models based on the model outlined so far in this section. In the remainder of this section we shall describe the transformation from Equation 9 to a function based on the term-based representation outlined in section 2.1. Specifically the discussion will be based on the probabilistic model known as the Binary Independence Model, as this is the most traditional variant of the overall probabilistic approach. This model was one of the first probabilistic models of IR, and will be used as an example of how the theoretical model is transformed into an actual retrieval model.

Before converting Equation 9 into an equation that can be estimated based on the probability of relevance and non-relevance of the terms in document *x*, it is necessary to consider how the probabilities of relevance and non-relevance interact. In particular, two aspects of retrieval are important: the independence of terms and what information is used to order documents.

The probabilistic model assumes that terms are distributed independently of other terms, that is the probability of seeing term *t* in a document is not affected by seeing term *s* in the same document. This is a simplifying assumption that reduces the computational complexity of the model. However it is necessary to define over what sets the independence holds. Two versions of the *independence assumption* were proposed in [RSJ76]. Both term independence assumptions assume that terms, query terms in particular, are distributed independently in the set of relevant documents: the probability of a term appearing in the relevant documents is not dependent on the probabilities of other terms appearing in the relevant documents. The two assumptions differ in whether the relevant document set should be distinguished from the whole document collection or only from the set of non-relevant documents.

"*Independence assumption I1*: The distribution of terms in relevant documents is independent and their distribution in all documents is independent"

"*Independence assumption I2*: The distribution of terms in relevant documents is independent and their distribution in irrelevant[7] documents is independent"

These two versions of the independence assumption are important in distinguishing whether we should measure the difference in the probability of a term's occurrence against the non-relevant documents (*I2*) or against its probability of occurrence the collection as a whole (*I1*).

The probabilistic model ranks documents according to their probability of being relevant to a query - the *ordering principle*. Two versions of this principle distinguish between the case where this

[7] The labels *irrelevant* and *non-relevant* are treated as synonymous in this paper.

probability is estimated based only on the *presence* of query terms within a document or the presence *and* absence of terms.

"*Ordering principle O1*: That probable relevance is based on the presence of search terms in documents"

"*Ordering principle O2*: That probable relevance is based both on the presence of search terms in documents *and* their absence from documents"

Four weighting schemes, F_1-F_4, can be derived from the combination of the two variants of the independence assumption and the ordering principle, Table 1.

	Independence assumption *I1*	Independence assumption *I2*
Ordering principle *O1*	**F_1**	**F_2**
Ordering principle *O2*	**F_3**	**F_4**

Table 1: Term weighting functions derived from the combination of independence assumptions and ordering principles

In [RSJ76] each of these possible strategies was instantiated to give an actual method for weighting a query term, summarised in Figure 5. The weighting methods themselves are based on a contingency table, Table 2, which converts the probability values into values that can be calculated from term occurrence information.

	rel	$\overline{rel}$	
$x_i = 1$	*r*	*n-r*	*n*
$x_i = 0$	*R-r*	*N-n-R+r*	*N-n*
	R	*N-R*	

Table 2: Contingency table to calculate term weights

where r = the number of relevant documents containing term x_i
n = the number of documents containing term x_i
R = the number of relevant documents for query q
N = the number of documents in the collection

Each of the four term weighting functions is a ratio of two proportions[8]:

- F_1 is the ratio of the proportion of relevant documents in which the query term *t* occurs (*ordering principle O1*) to the proportion of all documents in which *t* occurs (*independence assumption I1*).

- F_2 is the ratio of the proportion of relevant documents in which the query term *t* occurs (*ordering principle O1*)) to the proportion of all non-relevant documents in which *t* occurs (*independence assumption I2*).

F_3 and F_4 both use odds

- F_3, the ratio of 'relevance odds' (the ratio of relevant documents containing term *t* and relevant documents not containing *t* - *ordering principle O2*) and 'collection odds' (the ratio of documents containing *t* and documents not containing *t* - *independence assumption I1*).

- F_4 is the ratio of 'relevance odds' - *ordering principle O2* and 'non-relevance odds' (the ratio of non-relevant documents containing *t* and the non-relevant documents not containing *t* - *independence assumption I2*).

[8]It may be the case, especially when using small samples, that some of the values in the weights could be zero, resulting in error when taking logs. The solution is to add 0.5 to each cell in the numerator and denominator of each function.

$$w_{x_i} = \log \frac{P_q(x_i \mid rel)}{P_q(x_i)} = \log \frac{(r/R)}{(n/N)}$$

F1

$$w_{x_i} = \log \frac{P_q(x_i \mid rel) P_q(rel)}{P_q(x_i \mid \overline{rel}) P_q(\overline{rel})} = \log \frac{(r/R)}{((n-r)/(N-R))}$$

F2

$$w_{x_i} = \log \frac{P_q(x_i \mid rel) / P_q(\overline{x_i} \mid rel))}{P(x_i)/(P(\overline{x_i})} = \log \frac{r/(R-r)}{n/(N-n)}$$

F3

$$w_{x_i} = \log \frac{P_q(x_i \mid rel) / P_q(\overline{x_i} \mid rel)}{P_q(x_i \mid \overline{rel}) / P_q(\overline{x_i} \mid \overline{rel})} = \log \frac{r/(R-r)}{(n-r)/(N-n-R+r)}$$

F4

Figure 5: Term weighting functions F_1 - F_4

In [RSJ76], Robertson and Sparck Jones used the four term weighting schemes to carry out two sets of experiments. The first set was based on *retrospective weighting*. This involves deriving optimal weights to retrieve the relevant documents already found – the *known relevant set*. The second group of experiments were based on *predictive weighting*. Predictive weighting uses the weights from the retrospective stage to retrieve new documents. If the known relevant set is a representative sample of all relevant documents, then predictive weighting should be better at retrieving unseen relevant documents than the original term weights. Naturally, it is the latter, predictive, case that is mainly of interest as RF is intended to retrieve relevant documents that the user has not yet seen.

All functions outperformed no relevance weighting, and the *idf* function. F_1 and F_2, and F_3 and F_4 perform within the same range with F_3 and F_4 outperforming F_1 and F_2, and F_4 slightly outperforming F_3. This confirms Robertson and Sparck Jones' intuition that ordering principles *O2* is correct and that it is necessary to consider both presence and absence of query terms. No conclusive evidence was provided to distinguish between the two versions of the independence assumption, however Robertson and Sparck Jones favoured the second, *I2*, assumption as the more realistic assumption.

Given that the preferred weighting scheme is F_4, the odds function in Figure 6 (Equation 10**a**) can be converted to that of Equation 10**b** by eliminating the division operators. By noting that $P_q(\overline{x_i} \mid rel) = 1 - P_q(x_i \mid rel)$, and $P_q(\overline{x_i} \mid \overline{rel}) = 1 - P_q(x_i \mid \overline{rel})$ it is possible to convert the representation of F_4 in Figure 6 to that in Equation 10**c**.

$$w_{x_i} = \log \frac{P_q(x_i \mid rel) / P_q(\overline{x_i} \mid rel)}{P_q(x_i \mid \overline{rel}) / P_q(\overline{x_i} \mid \overline{rel})} = \log \frac{P_q(x_i \mid rel) P_q(\overline{x_i} \mid \overline{rel})}{P_q(x_i \mid \overline{rel}) P_q(\overline{x_i} \mid rel)} = \log \frac{P_q(x_i \mid rel)(1 - P_q(x_i \mid \overline{rel}))}{P_q(x_i \mid \overline{rel})(1 - P_q(x_i \mid rel))}$$

a **b** **c**

Equation 10: Term weighting function based on term's distribution in relevant and non-relevant documents
where w_{x_i} = the weight of term x_i

This equation (Equation 10**c**), which expresses the F_4 function solely as a factor of the *presence* of a term in the relevant and non-relevant documents, can alternatively be represented as in Equation 11. The probability of relevance of a document, then, is measured as the sum of the term weights of the query terms in the document, i.e. the sum of the F_4 weights of each query term in the document.

$$w_{x_i} = \log \frac{p_i(1-q_i)}{q_i(1-p_i)}$$

Equation 11: Term weighting function based on term's distribution in relevant and non-relevant documents
where w_{x_i} = the weight of term x_i, $p_i = P_q(x_i|rel)$ and $q_i = P_q(x_i|\overline{rel})$

The function in Equation 11 was examined as a basis for ranking terms for query expansion. Robertson, [Rob90], argued that a weighting function that ranks terms for *matching* (as in Equation 10) may not be appropriate for term *selection*[9]. That is, the degree to which a term indicates relevant material (matching) is not *necessarily* related to how well a term will improve retrieval effectiveness if added to a query (term selection). For term selection, Robertson proposed the formula in Equation 12, which provides a better estimate for how much a term will increase a search's effectiveness. Terms should be chosen for expansion based on the value shown in Equation 12 rather than the *w* value from Equation 11. Equation 12 incorporates the *w* value of a term but also takes into account the different between the relevant and non-relevant distributions based on *i*.

$$a_i = w_i(p_i - q_i)$$

Equation 12: Formula for ranking expansion terms based on term *t*'s distribution in relevant and non-relevant documents
where a_i = the value of term *i* for query expansion, w_i = weight of term *i* given by Equation 11, $p_t = P_q(x_i \mid rel)$ and $q_i = P_q(x_i \mid \overline{rel})$

The formula in Equation 12, with the appropriate substitutions for p_i and q_i becomes the term ranking function in Equation 13. This allows the calculation of Equation 12 based on the distribution of terms within the relevant documents and the collection. It should be made clear here that, although at each iteration of RF the same calculations are taking place (the weighting functions are identical even if that values are not), theoretically different probabilities are being calculated at each iteration: the distribution that calculates $P_q(rel \mid x)$ and $P_q(\overline{rel} \mid x)$ are different at each iteration [VR86].

$$w_i = \log \frac{r_i / (R - r_i)}{(n_i - r_i) / (N - n_i - R + r_i)} \bullet \left(\frac{r_i}{R} - \frac{n_i - r_i}{N - R} \right)$$

Equation 13: Term expansion ranking function
where r_i = the number of relevant documents containing term *i*
n_i = the number of documents containing term *i*
R = the number of relevant documents for query *q*
N = the number of documents in the collection

The F_4 reweighting function calculates weights for terms based on their distribution in the relevant and non-relevant documents. The probabilistic model is then a retrieval model that is specifically designed for RF. At the start of a search, of course, there is no relevance information to estimate the probabilities in Equation 10. One standard solution to this problem is to use a weighting function that does not depend on relevance information, such as *idf*. After an initial ranking of documents and relevant information has been obtained, a function such as F_4 can be used to provide improved term weights. The use of *idf* comes from substitution of appropriate values for *r*, *R*, and *n* into the F_4 weight in Figure 6.

It is possible to treat the query as an additional, and relevant, document and use the F_4 weight, however this will turn into something very like an *idf* weight [RWH+93]. An alternative to this was proposed by

[9] In [Rob86] Robertson also discussed the appropriateness of the 0.5 addition to the entries in the F_4 calculation, arguing that better estimations are more suitable for selecting new query terms.

Croft and Harper [CH79] based on the formula in Equation 8. This approach ranks documents by a function such as *idf*, assumes the top *n* documents are relevant, then uses these so-called *pseudo*-relevance assessments to estimate values for p_i and q_i in Equation 11. This will be discussed more fully in section 3.5.

This fundamental approach to probabilistic modelling has been extended in many ways, in particular to incorporate within-document frequency information [RW94]. Pertinent additions or modifications will be described, where appropriate, in later sections of this paper. An historical overview of the probabilistic model can be found in [SSJ+00a, SSJ+00b].

2.2.4 Logical model

In [Mar64], Maron hinted a potentially useful difference between the Boolean logic exact-match process and the process of logical implication. This difference distinguishes between the Boolean *matching* of text representations, in which the system is restricted to an exact formula, and the *inference* of information needs, by which process the system can infer more about what may be relevant than is stated in the query.

The advantages of implication or inference as the basis for a retrieval algorithm are demonstrated in the *logical modelling* approach to retrieval. This class of models originates from a proposal by Van Rijsbergen [VR86] that relevance can be modelled as a process of *uncertain inference*. More precisely the relevance of a document representation can be measured by the probability that the information in a document *infers* the information in a query[10], Equation 14.

$$P(d \rightarrow q)$$

Equation 14: Relevance measured as uncertain inference

This view was encapsulated in the logical uncertainty principle, [VR86]:

> "Given any two sentences x and y; a measure of the uncertainty of y $\rightarrow$ x related to a given data set is determined by the minimal extent to which we have to add information to the data set, to establish the truth of y $\rightarrow$ x."

That is if the information in a document, *d*, does not infer the information in a query *q* how much would *d* have to be changed to be relevant to *q*? The degree of necessary change to *d* allows the calculation of the probability of the inference.

As a simple example, if the query is about *animals* and a document mentions *dogs*, *ponies*, *cats*, but does not explicitly mention *animals,* then the document would not be retrieved by standard term-matching retrieval algorithms. By including information that *dogs*, *ponies*, and *cats* are kinds of *animals*, then it can be asserted that the document may be relevant and should be retrieved. Such an approach was taken by Lalmas, [Lal96], who used ontological relationships to express how many *transformations* or substitutions of this type would be necessary before a document's content inferred a query. In Lalmas's model, the number of substitutions gave a measure of the uncertainty associated with the inference.

The core logical models are based on non-classical logics as the classical notion of inference has several undesirable properties for retrieval, e.g. in classical logic the inference, $d \rightarrow q$, would hold if *d* did not contain any information, and the majority of logical models of IR are based on a possible worlds semantics, in which each possible world represents a possible combination of events. One possible representation is one in which a possible world represents a possible combination of terms. For example, given a set of indexing terms $\{t_1, t_2, t_3, ..., t_{10}\}$, there would be 2^{10} worlds: a world in which all terms are true, one in which all terms except t_1 is true, one in which all terms except t_1 and t_2 are true, and so on. In this representation each document and the query is associated with a world. The similarity of a document to the query is given by the *distance* between the document world and the query world[11].

---

[10] This is the most common version of the principle. Some authors have tried modelling the inverse; the degree to which the information in the query infers the information in the document $P(q \rightarrow d)$, or a combination of both measures, e.g. [Nie89]

[11] This assumes the Closed World Assumption, i.e. any fact not known to be true is assumed false.

Consider the example below, Figure 6, containing two documents indexed by a number of terms drawn from the set of indexing terms $\{t_1, t_2, t_3, t_4, t_5\}$. d_1 is indexed by the conjunction of terms t_1 and t_2, d_2 is indexed by the conjunction of terms t_1, t_2 and t_3, and a query, q, indexed by $t_1$and t_5.[12]

$d_1 = <1, 1, 0, 0, 0>$ $d_2 = <1, 1, 1, 0, 0>$ $q = <1, 0, 0, 0, 1>$

Figure 6: Possible worlds representation of d_1, d_2 and q

A simple retrieval model can be defined by asserting that all worlds (documents) have a distance of 1 from a query, q, if the intersection between the world and q is non-empty and the distance is 0 if the intersection is empty. This model would retrieve both d_1 and d_2 for q and corresponds to a Boolean disjunction of query terms. A Boolean conjunction of terms would be modelled by requiring the intersection of a world w and q to be identical to q.

Replacing the 1 and 0 in Figure 6 by term weights, such as *idf* or *tf*, gives the representation used by the vector-space and probabilistic models described previously. The distance between the query and document worlds is given by the similarity or probability functions described before. Thus the logical model can be used to encapsulate the three retrieval models outlined previously, see [Hui96].

As in the example above, the principle of transforming documents and queries can be used to incorporate semantic information into the retrieval process. For example, consider a query t_2, and information that t_2 is a synonym of t_3 (from a thesaurus or dictionary). We can then assert that both d_1 and d_2 should both be retrieved, but that d_2 should be retrieved first as it undergoes fewer transformations than d_1 to be relevant. We can also use representations based on different transformation principles, definitions of similarities, or definitions of possible worlds to give different retrieval models. [LaBr98] give a more detailed introduction to logical modelling of IR.

These models have the potential to be the very powerful models in IR as they attempt to model the *semantics* of information and can incorporate, within a single framework, retrieval tools such as thesauri. In addition, they also allow for multiple relations to hold – they can be used to specify *which* relations cause relevance (see [VR86]). The formal nature of logical models mean that they also allow for formal *comparisons* between IR systems, e.g. [Hui96]. Crestani et al, [CLVR98], give an overview of current models and approaches in logic-based information retrieval.

RF has, so far, not been a major concern of existing logical models but it is possible to imagine several approaches to the problem. We shall describe these based on the following example of a concept based on an example given in [Seb94] which describes the class of documents which appeared in the proceedings of *SIGIR93*, whose author is a member of the institution *IEI-CNR* and which deal with *logic,* Figure 7.

(**and** paper
(**func** appears-in (**sing** *SIGIR93*))
(**all** author (**func** affiliation (**sing** *IEI-CNR*)))
(**c-some** deals-with *logic*))

Figure 7: Terminological representation of a concept
Bold type indicates features of the representation language.

i. content modification. This approach is the most similar to that taken by the statistical RF models described previously. Here, the content of query is modified, e.g. by adding or deleting terms, or perhaps by altering connectives. For example in the above example we could refine the query to retrieve only those papers that deal with *modal_logic*. This would retrieve only concepts that specifically mentioned *modal_logic,* Figure 8, rather than the more general concept *logic*.

[12]Where 1 signifies that the proposition term t indexes the document is true, 0 signifies that the proposition is false.

(**and** paper
(**func** appears-in (**sing** *SIGIR93*))
(**all** author (**func** affiliation (**sing** *IEI-CNR*)))
(**c-some** deals-with *modal_logic*))

Figure 8: Terminological representation of a concept regarding *modal_logic*

or broaden the query by omitting one of the conditions, e.g. to retrieve all documents about logic written by a member of *IEI-CNR*, irrespective of where the paper was published. This would be a matching on only some of the components of our concept, as shown in Figure 9.

(**and** paper
(**all** author (**func** affiliation (**sing** *IEI-CNR*)))
(**c-some** deals-with *logic*))

Figure 9: Terminological representation of a concept

ii. personaliation of concepts. In addition to modifying the content of the query we could incorporate personalised thesaural knowledge. In the example, the term *logic* need not refer to a single term but could refer to a class of terms, e.g. *modal_logic*, *conceptual_graphs*, *cumulative_logic*, etc. This knowledge can be used as default values in retrieval but we could tailor this information to individual users based on feedback information. That is, the system automatically learns important synonymous concepts for individual users.

iii. uncertainty modelling. Logical concepts and rules reflecting thesaural knowledge are often associated with uncertainty values such as probabilities to reflect the importance of concepts or strength of relationship between concepts. These values can be changed in a similar fashion to the vector-space or probabilistic models to reflect important concepts in a search or the strength of association between concepts. Based on the example concept in Figure 8, for example, we could change the query to treat the author's affiliation as more important than the topic of the paper.

iii. rule modification or refinement. In this case, the information given by analysing the relevant documents is not only used to expand the query as in traditional feedback but is also used to modify the rules of the system. Examples of this approach include systems to select rules for retrieving documents, e.g. [DBM97] and the use of abductive logic to create new rules for retrieving documents, [Mull98].

2.3 Presentation of retrieved documents

A lengthy discussion of interfaces to IR systems will not be given at this point. Unless otherwise stated we shall assume that retrieved documents are presented either as a list (best-match) or set (exact-match). Hearst [Hea99] discusses the wide range of graphical and visualisation techniques that have been suggested for IR systems. Interfaces designed specifically for RF will be discussed in more detail in section 6.

2.4 Evaluation of retrieval systems and relevance feedback

We will now discuss the evaluation of IR systems and RF. The most common evaluation tool for IR systems is a *test collection*. This is a set of documents, a set of queries and a list of which documents are considered relevant for each query. The list of documents assessed as being relevant for each query – the *relevance assessments* – is usually not gathered from real-life search data. Rather test collections are usually constructed within a laboratory setting. Currently the foremost example of test collection construction is to be found within the TREC (Text REtrieval Conference) initiative, [VH96].

Test collections are primarily used for comparative evaluation: comparing the performance of two systems, or two versions of the same system on the same set of queries. Two standard evaluation measures are commonly used with test collections: *precision* and *recall*. Recall is measured as the ratio of relevant documents retrieved to the number of relevant documents in the collection. Precision is the ratio of relevant documents retrieved to the number of documents retrieved. In a best-match, or ranking model, recall and precision figures can be calculated at various points in the document ranking to give an indication of performance at different levels of retrieval. Typically this would be done at 10% recall,

i.e. 10% of relevant documents retrieved, 20% recall, 30% recall, etc. to give a set of 10 recall-precision figures), Figure 10.

Recall	Precision
10	67.3
20	65.9
30	59.2
40	45.3
50	36.7
60	33.3
70	21.9
80	19.7
90	15.3
100	12.1
average precision	**37.67**

Figure 10: Example recall and precision figures

With a test collection, the recall-precision (RP) figures for each query are averaged to form a single set of recall-precision figures[13]. The averaged RP figures are often averaged across the recall points to give a single value – the *average precision* value, Figure 10.

RP figures are often represented graphically. Figure 11 shows an example of a recall-precision graph drawn from the RP figures of two systems on the same test collection. As the line for System 1 is entirely above the line for System 2 we can infer that System 1 is better than System 2.

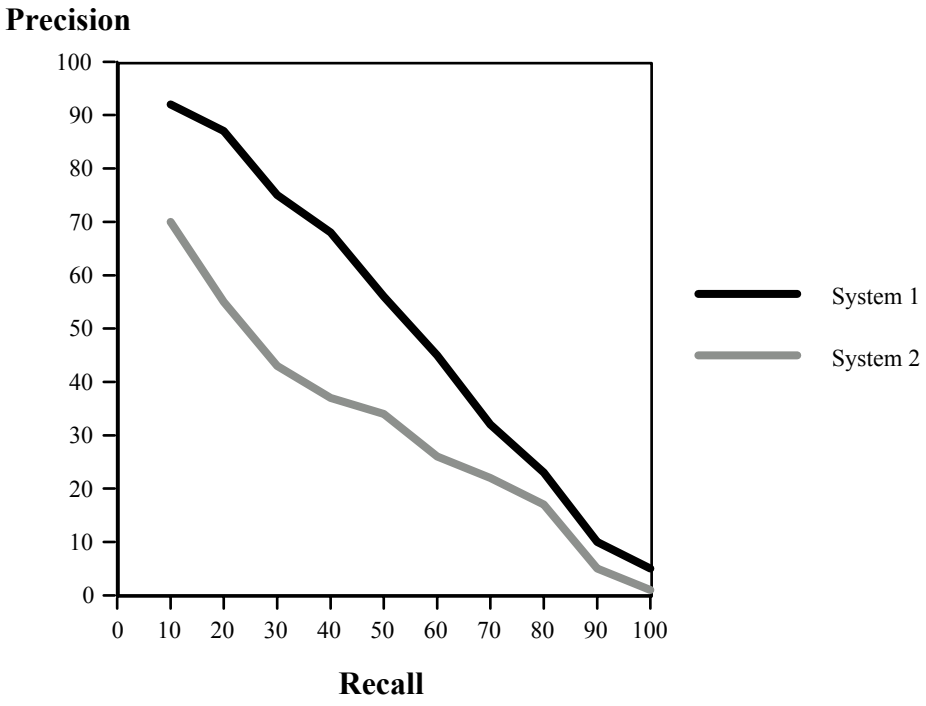

Figure 11: Example RP graphs

[13]Interpolation measures are necessary for queries whose recall levels differ from the standard, e.g. the example in Figure 10 is based on 10 recall levels, any query with a number of relevant documents different from a multiple of ten. Interpolation is often used to calculate a 0% recall figure to give an 11pt recall-precision table.

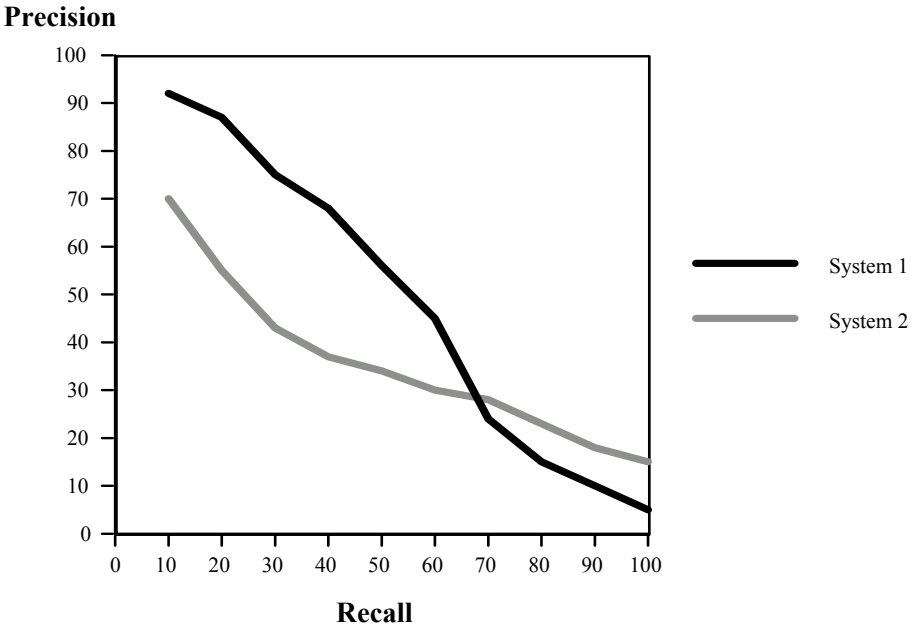

Figure 12: Example recall-precision graph

Figure 12 shows the results of the two systems for a different test collection. In Figure 12, the two lines cross at 70% recall, so we can say that, on the average of the queries tested, System 1 was better than System 2 at high recall levels (initially better at retrieving the relevant documents). On the other hand System 2 was better at lower recall levels (if the user is looking for *all* the relevant documents they will find them first with System 2).

Although these measures have been widely criticised for being capable of misrepresentation [FMS91], not reflecting the dynamic, situational and subjective nature of information seeking [BI97], and not reflecting *users'* evaluation criteria, e.g. [Su94], they have remained popular and standard measures of assessing an IR system performance.

However, as early as the early 1970's Chang et al., [CCR71], demonstrated that evaluation of RF algorithms poses certain problems for recall and precision. Given that RF, as described here, attempts to improve recall and precision by using information in marked relevant documents, it is usually the case that one of the main effects of RF is to push the known[14] relevant documents to the top of the document ranking. This *ranking effect*, will artificially improve RP figures for the new document ranking simply by re-ranking the known relevant documents. What is not directly tested is how good the RF technique is as improving retrieval of *unseen* relevant documents – the *feedback effect*. Chang et al [CCR71] investigated three alternatives, originally suggested by Ide and briefly outlined here to measure the effect of feedback on the unseen relevant documents:

- *residual ranking*: in this technique, the documents which are used in RF are removed from the collection before evaluation. This will include the relevant and some non-relevant documents. After RF, the RP figures are calculated on the remaining (*residual*) collection. The advantage of this method is that it only considers the effect of feedback on the unseen relevant documents but the main disadvantage is that the feedback results are not comparable with the original ranking. This is because the residual collection has fewer documents, and fewer relevant documents, than the original collection.

  A further difficulty is that, at each successive iteration of feedback, RP figures may be based on different numbers of queries. This arises because relevant documents are removed from the collection. If all the relevant documents are removed for a query, then this query cannot be used in subsequent iterations of feedback as there are no relevant documents upon which to calculate

[14]These are the relevant documents that are used for RF.

recall-precision figures. This method of evaluation is, then, biased somewhat towards queries that have more relevance assessments or those that perform poorly during initial iterations. An alternative, e.g. [SB90], is to only use the residual collection of both the rankings before and after feedback. This means that the two rankings are directly comparable but this method is really only suitable for small numbers of feedback iterations, otherwise the number of relevant documents in the residual collection can become relatively small and unrepresentative of the entire set of relevant documents.

- *freezing*. The method known as freezing is based on the rank position of documents and comes in two forms: *full freezing* and *modified freezing*. In full freezing the rank positions of the top *n* documents, the ones used to modify the query, and are frozen. The remaining documents are re-ranked and RP figures are calculated over the whole ranking. As the only documents to change rank position are those below *n* (the ones used for RF) any change in RP happens as a result of the change of rank position of the unseen relevant documents. There is, then, no ranking effect. In modified freezing, the rank positions are frozen at the rank position of the last marked relevant document.

  The disadvantage of freezing approaches is that at each successive iteration of feedback a higher proportion of relevant documents are frozen. This means that the frozen section of the ranking contributes more to recall-precision at later iterations of RF, so although RF may work better at these later iterations, it can appear to be performing more poorly due to the higher contribution of the frozen documents.

  In the previous discussion on the residual method of evaluating feedback runs, we mentioned that the residual collection method was forced to eliminate queries once all the relevant documents had been found. For the freezing methods, once all the relevant documents have been found for a query, recall-precision figures can still be calculated. However the recall-precision figures will not change once all the relevant documents have been frozen. Intuitively this seems correct: once we have found all the relevant documents for a query, feedback does not improve or worsen retrieval effectiveness.

- *test and control groups*. In this technique, the document collection is randomly split into two collections - the test group and the control group. Query modification is performed by RF on the test group and the new query is then run against the control group. RP is performed only on the control group, so there is no ranking effect. Successive queries can be run against the control group to assess modified queries on what can be regarded as a complete document collection unlike the residual ranking method. Unlike the freezing methods, all relevant documents in the control group are free to move within the document ranking. This means that recall-precision figures, before and after query modification, are directly comparable.

  The difficulty with this evaluation method is splitting the collection. It is easy to randomly split a document collection (e.g. by putting all evenly numbered documents in test group and all odd numbered documents in the control group). However, a random split will not ensure that the relevant documents are evenly split between the two collections. Neither will it ensure that the relevant documents in the test group are representative of those in the control group. Other factors such as document length or distribution of index terms may also be important to the RF method being tested, and may not be equally split between the two collections.

Each of these methods has advantages and disadvantages but all are standard methods of assessing RF algorithms. However, they only compare the performance of the algorithms in an idealised setting. For example, it is usual to use the same number of documents per feedback iteration to modify the query. A user, however, is unlikely to examine an identical number of documents per search iteration. Also RF experiments based on recall-precision assume complete knowledge of the document collection: a fixed set of relevant documents is known beforehand. In interactive searching this is also unrealistic as what a user finds relevant may change over time, e.g. [Kuh93, Ell89, SW99, Vak00a]. Additional methods are required to test the effectiveness of RF algorithms in more realistic settings.

A final point regarding these measures of RF evaluation is that they may not be directly comparable: each measure may appear to give different results depending on how the results are compared and on what factors affect the retrieval. An example of this is given in Table 3 which shows the results of RF

on the same collection[15] but evaluated using the three RF evaluation schemes. An initial document ranking, for each query, was obtained using the *idf* weighting function, followed by four iterations of RF, in which the top 6 expansion terms were added, based on an F_4 ranking of expansion terms. 50 new documents were used in each iteration of feedback. After feedback all query terms were weighted using the *idf* weighting scheme and these values were used to score documents. Table 3 gives the percentage change, over no feedback, after four iterations of feedback using each of the three evaluation techniques.

AP 88	Full freezing	Residual collection (removal)	Residual collection (no removal)	Test and control
%age increase over no feedback	+2.9%	-77.0%	-25.0%	+21.5%

Table 3: Example RF evaluation

As can be seen from Table 3, the results vary according to how they describe the retrieval effectiveness of the system. Full freezing (column 2) gives a small increase in the effectiveness of the system. The test and control method gives a larger percentage increase in effectiveness (column 5). These two approaches give different absolute performance figures (average precision) as they use different data to calculate *idf* values, F4 values and do not have identical terms in the collection. The test and control method used two less queries (as all the relevant documents for this query appeared in the test collection), and several of the queries were expanded by terms that appeared in the test collection but not the control collection[16]. These differences cause the different performance figures for the two evaluation methods.

The residual collection method (column 3) gives a large drop in retrieval effectiveness. This is because the residual collection method eliminates queries that have no relevant documents in the residual section of the collection. This means that queries, for which all relevant documents have been retrieved in early iterations of feedback, have been removed from the evaluation. The queries that are being used to calculate average precision are the ones for which the system finds it difficult to retrieve the remaining relevant documents[17]. If we do not remove queries when all relevant documents are found and, instead use the RP figures from the previous iteration, then we obtain the figure in column 4 for residual collection. This is an attempt to soften the effect of removing queries that perform well. This also shows a drop in retrieval effectiveness but not so severe a drop as in column 3. The drop in retrieval effectiveness is caused, again, by the effect of the queries for which the system finds it difficult to retrieve all relevant documents.

An alternative method of examining RF performance is to plot the average precision values at each iteration of feedback, as in Figure 13. We can see that different methods give different shaped graphs. The freezing graph gives slight, but steady, increases in retrieval effectiveness at each iteration of feedback. The test and control method gives an initial large increase followed by decreases at the last iteration of feedback. The residual methods, however, give very different, but similar-shaped graphs: large decrease initially followed by increases in performance at later iterations.

The graphs can be used to highlight interesting areas – where RF is working well or where it is operating poorly. However as with recall and precision the graphs can be misleading: all four lines plotted in Figure 13 are evaluating the same feedback technique on the same collection. The point is that the evaluation measures are calculating different aspects of feedback: freezing is measuring *cumulative* effectiveness, residual collection is measuring the effectiveness of retrieving *only* the remaining relevant documents and test and control is measuring the relative performance of the modified queries produced at each iteration.

[15] AP (Associated Press) collection 1988.

[16] This was also true for one of the original query terms.

[17] The remaining queries may also include some queries that have a large number of relevant documents, but this is unlikely to be the case in this test as 200 documents have been used for feedback whereas the queries have an average of only 35 relevant documents per query.

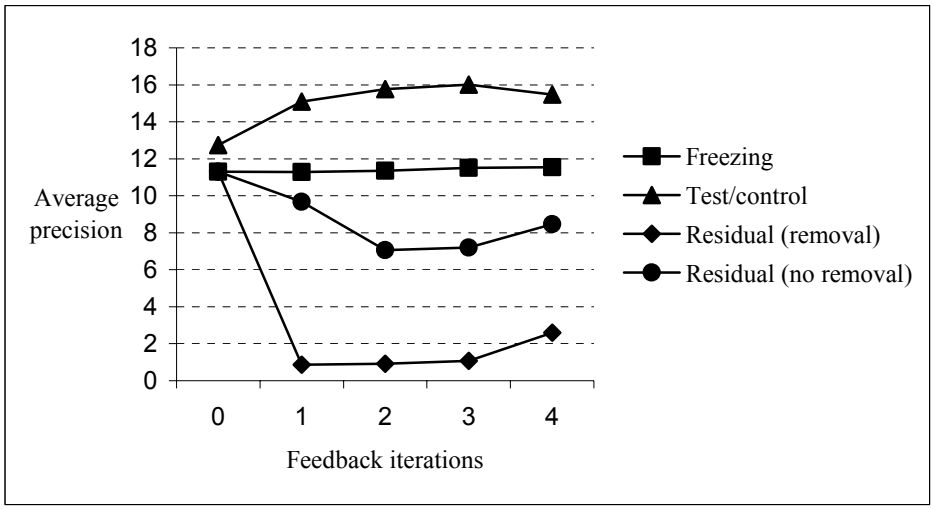

Figure 13: Average precision over 4 iterations of feedback

2.5 Summary of RF

In this section we shall summarise the main points from the previous sections and outline some of the major issues in the core RF models. In section 2.5.1 we shall summarise the comparison between Boolean and best-match models, in section 2.5.2 we shall compare the types of best-match model, and in section 2.5.3 we shall compare the two main components of RF – query term reweighting and query expansion.

2.5.1 Boolean vs Best-match

Although Boolean models are still popular and have strong advocates, e.g. [FST+99], in general there are many advantages to best-match models over exact-match models. The first advantage is that the user does not need to generate a query expression in the same way as with the Boolean model. Instead they can enter a natural language expression. This means that users can initiate retrieval sessions without knowledge of the collection, previous searching experience or experience in creating Boolean queries.

A second difference is that ranking documents allows the users to interact in a more meaningful fashion with the system, [Beau97]; documents are presented in order of match and documents are not excluded if they miss out elements of the query.

Thirdly the system can automatically alter a query through RF. The main strength of best-match models is that they allow for *iterative* improvement, often using similar techniques to retrieve documents as to modify queries. The strength of ranking models for RF is that, after initial querying, the user can interact without further *describing* the information for which they are searching. The RF algorithms discussed in the main body of this paper deal almost exclusively with best-match algorithms. In the next section we shall look at the relative performance of the best-match models discussed previously.

2.5.2 Relative performance of best-match models

In [SB90] Salton and Buckley investigated the relative performance of 12 feedback algorithms on six standard test collections[18]. Several of the feedback algorithms (Ide-dec-hi, F_4, Rocchio, and three versions of Rocchio with scaling factors for query, relevant and non-relevant set) have already been discussed.

A further version of the Ide scheme was used, the *Ide-regular* scheme, [IdS71], which uses all retrieved, non-relevant documents. The Ide-regular is based on the Rocchio formula but omits the

[18] CACM, CISI, Cranfield, Inspec, MEDLARS and NPL collections. These are relatively short document collections ranging from 1, 033 documents (MEDLARS) to 12, 684 documents (INSPEC).

normalisation of the relevant and non-relevant documents by the number of relevant/non-relevant documents. Equation 15 shows the Ide-regular formula.

$$Q_1 = Q_0 + \sum_{i=1}^{n_1} R_i - \sum_{i=1}^{n_2} S_i$$

Equation 15: Ide-regular

Two of the other algorithms were modifications of F_4. The first used the ratio [Rob86] n_i/N to replace the 0.5 correction factor introduced to cope with the case where no relevant documents were retrieved ($R = 0$) or when no relevant documents contain an individual term ($r = 0$), Equation 16.

$$w_{x_i} = \log \frac{\left(r_i + {}^{n_i}\!/_{N}\right) \Big/ \left(R - r_i + 1\right)}{\left(n_i - r_i + {}^{n_i}\!/_{N}\right) \Big/ \left(N - n_i - R + r_i + 1\right)}$$

Equation 16: Modified F_4 function using n_i/N

The second modified F_4 scheme placed extra emphasis on terms that appeared in the query. Specifically this was achieved by assuming that a term's appearance in the query is equivalent to an occurrence in 3 relevant documents (i.e. $r_i = r_i + 3$, $R = R + 3$).

Salton and Buckley found that, for all collections, except the NPL collection[19], the models performed fairly consistently with respect to each other, with the Ide-dec-hi performing best overall. In general, although the probabilistic model performed well, it did not quite reach the performance level set by the vector space models. This was advantageous as the vector space Ide-dec-hi RF technique is computationally very efficient.

Salton and Buckley also provide some general guidelines based on predicting RF performance. For example, short queries, on the whole, do better with RF than longer queries. Longer queries, or those queries with more terms that appear in the relevant documents, will tend to achieve better initial rankings. This means that there is greater *potential* improvement to be gained from RF on short initial queries. For a similar reason queries that do poorly on initial runs tend to obtain greater improvements with RF than those with good initial retrieval runs

Finally, domain-specific collections also perform better with RF than domain-independent collections. This may be because it is easier to select good expansion terms from a domain-dependent collection, or because the ambiguity of search terms is less significant.

As well as considering variations on the probabilistic and vector space models Salton and Buckley investigated weighting document terms (as opposed to binary weighting based on term presence/absence in each document) and three variations on query expansion - no expansion (only reweighting), full expansion by all the terms in the relevant documents and partial expansion, adding only some of the relevant terms to the query. For all collections, again except the NPL, weighting document terms gives a considerable improvement in feedback, as does full expansion by all terms in the relevant set[20]. Queries should be expanded by those terms that appear with the highest frequency in the relevant documents rather than those with the highest feedback weight.

Rocchio's original formula and the Ide-dec-hi variant perform the joint function of modifying query terms and query term weights. These and the other vector space RF techniques use the original

[19]The NPL collection differed in a number of ways from the other collections investigated. It had much shorter query and document vectors, and lower term frequency. For this collection, although the same relative ordering was found between algorithms, binary document weighting was better than weighting document terms. This may result in the vector-space length normalisation procedure being ineffective for this collection.

[20]Although full expansion is preferable, partial expansion also gives good results and can be used to reduce storage. In larger collections than the ones tested here partial expansion may actually perform better than full expansion.

document term weights to calculate the new term weights for query terms. The probabilistic-based F_4 weights, on the other hand, are derived directly from the feedback process itself. The traditional probabilistic version presented in section 2.2.3 however, ignores the frequency with which a term appears in the query and in documents. This latter feature has been extended in [RW94]. Harman, [Har92b] section 2.5.3, and Salton and Buckley, [SB90], both showed that query expansion and query term reweighting are essential to RF.

Salton and Buckley's experiments were carried out in an experimental setting. In such a setting, especially with smaller test collections such as the CACM, Cranfield, and NPL, we can assume complete relevance information; that we know all the relevant documents for a query. However in a real information-seeking situation, users will not necessarily assess every retrieved document, often they may only assess a small number of documents, before trying RF. This could be significant as a standard assumption in operational systems is to assume all documents that are not explicitly marked relevant should be treated as non-relevant. Sparck Jones, [SJ79], ran a set of experiments to test how well the probabilistic F_4 weighting scheme performed with little relevance information and demonstrated that even very few relevance assessments, as few as one or two relevant documents can still improve a search over no term weighting.

2.5.3 Query expansion vs term reweighting

In [Har88, Har92b] Harman examined the relationship between query expansion and reweighting in the probabilistic model. As the original probabilistic model did not incorporate the addition of new terms to the query, it is important to make sure that best possible terms are added. One obvious solution is to add all terms in the relevant documents but Harman hypothesised that improved performance could be obtained by ranking these terms and adding only a number of them to the query. This raises two questions both examined in [Har88]: how to rank the terms, and how many terms to add to the query?

In [Har88] she examined six techniques for ranking terms, and demonstrated on the Cranfield 1400 test collection, that adding between 20 - 40 terms much improved performance over adding all terms with a peak at around 20 terms. The best technique for ranking the terms was one that combined *idf*-like information and frequency of term occurrences in relevant documents.

In [Har92b] she extended this work, on the same document collection, using a set of new algorithms for term ranking, and reinforced the suggestion of adding around 20 terms to the query[21]. She also explored the relationship between query expansion and term reweighting: query expansion *and* reweighting of query terms gave increased performance, with the major benefit coming from query expansion component rather than reweighting. [Har92b] also explored a number of alternative methods for ranking terms. The details of these new algorithms are not significant here but what is important to note is that, although the improvements of certain of these techniques were similar, the terms they added to the query we not identical. This means that different algorithms may present different documents to the user based on the same relevance assessments. One possible way to exploit this is to combine methods for RF as in section 3.4, an alternative is to allow the user to make the choice of which terms to add to the query, discussed in section 5.

In this section we have outlined basic operations of IR systems and how RF is implemented in the major retrieval models. In the remainder of this paper we shall discuss extensions to these models to incorporate aspects such as changing information needs, alternative models and uses of relevance feedback, section 3. We shall summarise the overall features of *automatic* RF in section 4 and turn to the interactive aspects of RF in sections 5 – 7.

3 Extensions to RF

The three sections that follow all extend, rather than challenge, the RF techniques discussed previously. In section 3.1 we outline approaches to incorporate relations between terms. In section 3.2 we describe how the fact that what a user finds relevant may change over time. In section 3.3 we discuss negative RF - users making feedback decision on what is *not* relevant to their needs. In section 3.4 we discuss

[21] Experiments carried out by Magennis and Van Rijsbergen [MvR97] indicate that the optimal number of expansion terms for a test collection can vary between collections and query sets. Ruthven et al. [RLVR01] showed that smaller-scale expansion, with more careful selection of expansion terms, can perform better than larger-scale expansion.

the combination of evidence in RF: combining multiple queries, retrieval algorithms or feedback algorithms, and in section 3.5 we discuss pseudo-RF: employing RF without the user's involvement.

3.1 Dependence between terms

The vector space and probabilistic models assume that terms are independent of each other, that is the presence of one term in a document does not alter the probability of seeing another term in the same document. Although this simplifying assumption has facilitated the construction of successful retrieval systems, it is not true. Words are related by use, for example in phrases, and their similarity of occurrence in documents can reflect underlying semantic relations between terms.

Incorporating information on *co-occurrence* patterns of terms in documents may improve retrieval effectiveness as indicated by the Association Hypothesis [VR79]:

> "If an index term is good at discriminating relevant from irrelevant documents then any closely associated index term is also likely to be good at this."

Author such as Spiegel and Bennet, [SB64], as early as 1964, suggested that *dependency* information of this kind may be used to choose further search terms for query expansion. Not all query expansion based on dependence information is used for RF, for example we could use dependency information to automatically expand *initial* queries in the absence of relevance information from the user. However three investigations of dependency information, with a RF connection, are outlined below.

Van Rijsbergen, Harper and Porter [VRHP81] proposed using a maximum spanning tree (MST) in which each node represents a term and each link represents the association or similarity between the two terms. The MST links each term to its most similar terms as measured by the association measure. The association measure used in [VRHP81] was the EMIM (Expected Mutual Information Measure) measure, based on the probability distribution of the two terms. The MST can be potentially be used in many ways to expand a query. In [VRHP81] the most similar terms to the query terms (the ones directly linked in the MST) are added to the query. The query and expansion terms in [VRHP81] are also reweighted by a weight based on the F_4 weight. On the whole, Van Rijsbergen et al. show that their term dependence approach behaves better than the F_4 term independence weighting scheme. They also demonstrate the relative robustness of the MST approach, in that although, the EMIM-based MST gives superior results, alternative association measures do not give significantly different results.

Smeaton and Van Rijsbergen [SVR83] investigate query expansion and term reweighting using term dependence. Their investigation centred around three methods of query expansion: the MST approach of Van Rijsbergen et al, a Nearest Neighbours (NN) approach (this added terms that were statistically most similar to a query term) and query expansion by a list of possible expansion terms from the relevant documents. The third technique, expansion with terms from relevant documents is similar to the term independence approaches outlined in section 2. The results from these experiments were largely negative. Query expansion via the MST generally degraded performance over the unexpanded query, as did expansion via the NN or expansion terms chosen from the relevant documents. One striking feature was that the performance degradation increased as the number of terms added to the query increased. Smeaton and Van Rijsbergen point to the difficulty in estimating probabilities as the main reason for this failure.

In [Bha92] Bhatia also presented a model of dependence trees for query expansion to incorporate user specific information. Bhatia suggests that the dependence tree approach can be improved by not only being more selective about which terms appear in the tree but by weighting the links between elements in the tree according to user preference. The claim is that although spanning trees can suggest expansion terms based on statistical similarity they do not suggest them based on *conceptual* similarity.

The solution presented is to elicit from the user what concepts are present in documents and how they relate to each (how similar or dissimilar they are). This can be used to develop a new spanning tree that more accurately reflects the user's personal constructs based on concepts rather than explicitly mentioned terms. A spanning, or dependence, tree would have to be constructed for each user but the argument is that it would better support the users searching and choice of terms.

An alternative approach to exploiting term dependency is *term clustering* - grouping sets of related terms with a view to selecting query expansion terms from these sets. This can be achieved without relevance information (using only statistical information on term similarity to choose expansion terms) or with relevance information (using a combination of collection dependent information and information from the relevant set to choose expansion terms). Both these methods will typically rely on term co-occurrence methods to generate clusters but the term co-occurrence methods used in the literature have generally not provided convincing results [PW91].

The methods for incorporating term dependence outlined in this section have not produced the increase in retrieval performance that may be expected. Partly this may be due to the computational limitations of calculating and storing dependence information. Although the term independence methods, such as the F_4 term weighting scheme, do not explicitly capture term dependence, they do implicitly capture some degree of term co-occurrence. That is, although the term independence methods do not calculate explicit values for co-occurrence, one would expect that the terms in the term expansion list would have a greater than average degree of term co-occurrence. This is because good discriminators of relevance are those terms that appear more frequently in the relevant than non-relevant documents. How to use this co-occurrence information successfully, and in a computationally efficient manner, remains an open research question.

3.2 The dynamic nature of information seeking

Implicit to much of the early work on RF is the assumption that users have a fixed information need: that the information for which they are searching does not change over the course of a search. Whilst this may be true in certain cases, evidence from a range of studies on information seeking, e.g. [Kuh93, Ell89, SW99], show that information needs should be regarded as transient, developing entities rather than a fixed request.

The techniques discussed previously modify queries based on the difference between relevant and non-relevant documents but they do not consider *when* a document was marked relevant: a document marked relevant at the start of a search contributes as much to RF as a document marked relevant at the current iteration. If we assume that user's information needs are static then this is correct. However if the user's need is developing or changing throughout the search, then documents that were assessed as being relevant early in the search may not be good examples of what the user *currently* regards as relevant. Campbell, in a series of papers on developing information needs, has addressed this issue through the notion of *Ostensive Relevance*, [Cam95, Cam99, CVR96].

The basic premise behind Ostensive Relevance, [Cam95], is that documents selected at the current iteration of RF are the best indicators of what the user finds relevant; documents assessed as relevant in previous iterations are decreasingly useful at describing a user's information need. Relevant documents, then, are not seen as a set of *equally* important documents but sets of documents of *varying* importance. In [CVR96] Campbell and Van Rijsbergen produce an extension to the probabilistic model of retrieval that incorporates an 'ageing' component to term weighting. When calculating the weight of a term this ageing component incorporates when the documents containing the term were assessed relevant: if the documents were marked relevant at an early stage in the search then the term receives a lower weight than if the document was assessed relevant in recent iterations. The ageing component can be tuned to differentiate more or less strongly between older and more recent documents. In [Cam99] a preliminary test of this approach indicated that ostensive weighting can improve searches in fewer search iterations than non-ostensive approaches. Ruthven et al. also showed ostensive weighting as being beneficial for query expansion [RLVR02b].

Standard RF techniques, such as Rocchio or F_4, will also adapt to changing information needs but they will require more evidence to do so as they will require an accumulation of new evidence to outweigh the old evidence. Campbell's ageing component reduces this mass of evidence required to shift a query towards the new information need.

Berger and Van Bommel, [BVB97], present a model with similar aims. Their work is specifically aimed at characterising the content of documents through hyper-indices: hypertext representations of document indexes, such as the one shown in Figure 14.

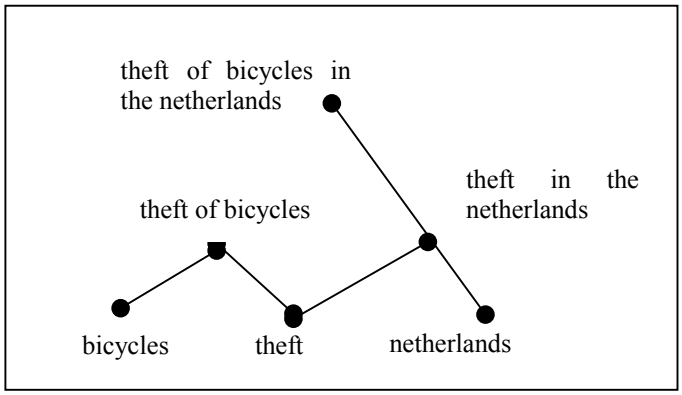

Figure 14: Hyper-index

Each node in the hyper-index corresponds to a potential query and is associated with a set of documents. The user can browse the hyper-index to select a query formulation for a search, and can move between documents and index descriptions at will. The nodes correspond to document descriptors: the more descriptors of a document that have been visited by a user, the more likely a document is to be relevant to the user. The more important a document descriptor is then the more it counts towards document retrieval and document ranking. The concept of descriptor importance is analogous to term weighting in the traditional document retrieval models presented in section 2. Relevance information is used to alter the importance of the document descriptors. In particular recency information is used to increase the importance of recently visited descriptors and lower the importance of descriptors visited earlier in the search.

Dynamic information needs also present a new problem for evaluation. If we assume a changing information need we can no longer rely on existing test collection methods as they also rely on the notion of a fixed information need. The assessment of recall in an interactive situation is especially problematic, as the desired set of relevant documents[22] will change from one search iteration to another. One further problem of RF evaluation in this context is what to measure: the quality of the feedback (how well does the system improve the user's query) or the quality of the adaptation to the information need (how well does the algorithm track how the query is changing)? These are not necessarily the same entity: potentially a RF algorithm could be good at describing the known relevant documents but poor at detecting how the user's relevance assessments are changing.

3.3 Negative RF

The majority of RF techniques are based on capturing the difference between the content of the relevant documents, those documents that the user has marked as containing relevant information, and the content of the non-relevant documents. The label 'non-relevant', however, if often used to refer to two groups of documents:

i. those that have been explicitly marked non-relevant by the user. In small test collections we can assume that the documents that have not been explicitly marked relevant by an assessor have nevertheless been assessed and judged non-relevant. In larger collections, such as those provided by the TREC initiative [Har93], a small set of documents is explicitly marked non-relevant, meaning they have been assessed as *not* containing relevant information.

ii. those that have not been assessed by the user. These documents may not have been retrieved or the user did not assess the documents, or the user implicitly rejected the document but did not provide an explicit relevance assessment. It is common to assume, for any query, that these documents will comprise the majority of the documents in the collection. The probabilistic model and vector space model both make use of this assumption, in that they do not differentiate between *assessed non-relevant* documents and *unassessed* documents. However some of these documents, if they had been viewed by a user, might have been assessed as relevant.

[22] That is the set of documents that the user would regard as being relevant if shown them at the current iteration, not the set of relevant documents used for feedback.

This section looks at using information from the former group of documents (**i.**) - those that have been explicitly marked as being non-relevant. This form of feedback is called *negative relevance feedback*. Negative relevance feedback has generally been regarded as problematic for three main reasons.

i. **Implementation**. One difficult issue of negative relevance feedback is that it is not clear how negative information should be handled by the system. A common decision in IR is to remove from the query those terms that have a negative weight – those terms that are better at retrieving non-relevant than relevant documents. Negative feedback can be used to better indicate which terms should receive a negative weight.

Belkin et al., in a long-running study of user's involvement in RF [BCK+96, BCC+97, BCC98, BCK+99], propose an alternative model. They hypothesise that terms which appear in negatively, as well as positively, assessed documents may be good query terms. These terms are good in the sense that they can retrieve relevant documents. However, these terms may appear in the wrong context in the document, or the document does not discuss them fully or discuss them in the way the user requires. In their model, what is important is not the distribution of a term between the relevant and non-relevant documents but the *context* of terms. Terms that appear only in negative documents can be used to indicate inappropriate contexts, main topics etc. of the useful terms and perhaps this could lead to the detection of different reasons for non-relevance.

Belkin and his colleagues carried out a series of experiments, mainly reported in [BCK+96, BCC+97], which examined how users utilitised negative feedback. In the experimental system reported in [BCK+96], subjects could explicitly mark a document relevant or non-relevant, and were given suggestions as to terms that could be added to the user's query. The terms themselves could be positively or negatively weighted. Although subjects preferred the system that allowed negative and positive feedback, they did not feel that negative feedback was very useful. Belkin et al. give several reasons for this, based on subject's comments. Often subjects were concerned that negative relevance assessments would stop the retrieval of relevant documents. That is, they were concerned that the system would not understand upon what information the negative decision was based. Similarly, subjects were concerned that negative RF would lower the rank position of relevant documents.

An additional concern for the subjects was that negative feedback was a more difficult decision to make. The experimental conditions, in particular the time constraint imposed by the experiments, led some subjects to feel that negative feedback was too unpredictable to use. Other reasons for the lack of use of negative RF include the perceived topic-dependency of negative RF, that is negative RF is only appropriate for some topics, the lack of control as to which terms were negatively weighted and problems relating to word stemming. This latter problem results from the fact that useful and non-useful terms may be stemmed to the same base stem.

One aim of negative feedback that was requested by users, also noted by Sumner et al [SYA+98], was the suppression of previously seen, non-relevant documents. These documents were discarded by the user but reappeared in the ranking if they matched the new query. A common request by users was that these documents were not re-retrieved.

The experimental results from [BCK+96] were equivocal but hinted at potential improvements when subjects used a mixture of negative and positive feedback. The experiments reported in [BCC+97] reiterated many of the conclusions from [BCK+96], namely that although users may use negative feedback, the gain in performance is not significant, and users are unsure about the process of making negative assessments. A more positive indication from [BCC+97] is that users' familiarity with negative feedback may be an important factor in its success: the more familiar a user is with this option, the more comfortable they are with using it.

ii. **Clarity**. It may also be difficult to specify under what conditions should a user should consider and mark a document non-relevant. There are many reasons why a document is not considered relevant, e.g. if the document contains absolutely no relevant information, contains no information related to what a user is searching for, contains topically related but non-relevant information, if the document is relevant but not relevant enough, and so on. Any of these definitions may apply within or across searches. The issue here is – when should a user mark a document non-relevant?

It could be argued that this problem also applied to positive feedback - when should a user mark a document relevant? However, we believe that this issue is more central to negative feedback for two

reasons. First, as indicated by Belkin et al.'s experiments, the effects of negative feedback are not clear to users. In a positive feedback situation, it is easier to see what *kind* of documents are being retrieved, and infer the change(s) that have been made by the system. The potential *harm* that a negative assessment may do to a search is not apparent because the user cannot see what documents have been suppressed by the feedback action.

Second, it may be the case that assessing non-relevance is a harder task than assessing relevance. That is, in practice, relevance and non-relevance are not opposite assessments. A user making a positive relevance assessment can often give detailed reasons for why a document is relevant, e.g. [BS98], but the reasons for non-relevance are likely to be based on what is lacking from the document, rather than what is present. The assessment of non-relevance, therefore, often requires reasoning about what is not contained within the document. An alternative to negative assessments, in this case, may be to use *partial* relevance assessments, e.g. [BI99]; rather than asking users to make binary, relevant or non-relevant, assessments on a document, the system allows the user to make a scalar or non-binary assessment of the document's relevance. We shall return to this point in section 7.3.

iii. Usability. The mechanisms for making relevance assessments are important. We shall discuss this in more detail in section 7.3.3 but a general point is that, even though RF techniques *can* improve a search, it is not always the case that users will make relevance assessments. Partly this may be due to a lack of awareness, on the part of the user, as to the function of RF; it may also derive from a fear of having an unknown effect on the search. The usability of making assessments can have an effect on how likely the users are to make assessments. It may be the case, for example, that the more complex the relevance assessment is, as a task, then the less likely users are to make more assessments. Similarly, if the process of making relevance assessments (operating the system) detracts from gaining relevant information (the task of using the system) then, again, the users may be less willing to explicitly assess documents. Asking users to spend time marking documents that are *not* relevant to their search may be difficult to achieve practically.

Dunlop [Dun97] presents a more specific argument against negative RF: namely that negative feedback, as implemented in the major models, is not only inconsistent across models but is often not performing the correct task. His paper is based on an intuitive view of what positive and negative RF *should* do. Namely, positive RF on a document at the top of the document ranking, or negative RF at the bottom of a ranking should have *little* effect on the query, as they both confirm the retrieval decision. In contrast, positive feedback on a document at the bottom of the ranking, or negative feedback on a document at the top of a ranking should have a greater effect on the query, as these feedback cases contradict the retrieval decision made by the system.

Dunlop compared this intuitive view against three models - vector space (using a modified Rocchio formula), probabilistic (F_4) and a query expansion technique (one in which negative RF reduces term weights). The data he used was not identical in all cases, so the results are not strictly comparable, however the general trends are important. He found that, in general, all models behave as expected for positive RF: the effect on the query is inversely proportional to how well the document matches the query. However for negative RF the systems differ. For the vector space implementation, the effect of negative RF on a poorly matching document is greater that on a highly matching document, although certain scaling factors can reduce this problem somewhat. The normalisation by document length in the vector-space model also means that the effect of negative RF is not reversible in this model.

The probabilistic model also does not behave intuitively for negative feedback, primarily because the F_4 scheme does not differentiate between documents that have been explicitly marked non-relevant and those that have simply not been marked relevant.

The third model - query expansion - will behave in line with Dunlop's intuitive view, if the system allows negative weights to be attached to terms, unlike most systems which will remove a term if its weight falls to zero or becomes negative.

Dunlop's investigation demonstrates the difficulty of incorporating negative assessments into RF. The further difficulty of incorrect contexts, identified by Belkin et al, remains a problem for positive *and* negative feedback. It maybe the case that keyword-based algorithms that we have examined so far require more complex mechanisms to make fine-grained analysis of keyword contexts for feedback.

3.4 Combination of evidence in RF

Many of the RF and retrieval techniques described so far have utilised a single query representation compared against a series of single document representations, using one retrieval algorithm. Many researchers have argued that better retrieval effectiveness may be gained by exploiting *multiple* query representations, retrieval algorithms or feedback techniques and *combining* the results of a varied set of techniques or representations. The combination of evidence from multiple sources is the topic of this section. In particular, we will highlight approaches to multiple query representation, section 3.4.1, multiple retrieval algorithms, section 3.4.2 and multiple feedback algorithms, section 3.4.3.

Before this, it is worth highlighting the two main arguments in favour of combination of evidence for IR and RF. Proponents of combining evidence, usually base their motivation on either *empirical* findings, or *theoretical* properties of evidence combination. The empirical evidence includes the fact that different retrieval functions or query representations will retrieve different documents, e.g. [Lee98]. A combination of query representations may increase the *recall* of a query, whereas the combination of retrieval functions may increase the *precision* of a search.

A strong theoretical basis for combining evidence was provided by Ingwersen, [Ing94, Ing96]. His research argues that multiple representations of the same object, for example a query, can provide better insight into the object than a single good representation. However, what is important is that the multiple sources of evidence must each provide not only a different viewpoint on the object, but that these viewpoints must have different cognitive bases. Here, more evidence alone is not better, what is important is the *variety* of evidence. This *intentional redundancy* – multiple descriptions of the same object – can help uncover information about the user. Multiple query representations, for example, can provide different interpretations of a user's underlying information need, or provide more detail about how the user is making relevance assessments.

3.4.1 Multiple query representations

Belkin et al., [BKF+95] differentiated between two types of retrieval combination based on multiple representations of a query:

i. **query combination.** In this case the scores for a document are computed directly from query-document scores, using the same retrieval engine but using different version of the query.

ii. **data fusion.** If different retrieval systems are used to compute query-document similarity scores then the scores may not be compatible for combining. For instance, the scores may be in a different range or the scores cannot be normalised to give comparable rankings. In this case it is necessary to combine evidence from the document *rankings* rather than document-query similarity scores. This form of evidence combination is known as *data fusion*.

Belkin et al. experimented on both kinds of combination, showing that data fusion generally performed less well than query combination approaches. The general trend of the experiments presented in [BKF+95] was that combination of query representations *can* improve retrieval effectiveness but that is difficult to determine what are good sources of evidence to combine. Ruthven et al., [RLVR02a], also showed similar results for retrieval using a variety of term weighting schemes. Both these experiments only looked at initial retrieval, with no RF.

Ruthven et al.'s experiments were extended in [RLVR02a] to the RF case where they showed that relevance information, the relevant documents, could be used to *select* which weighting schemes should be used to weight query terms. That is, it is possible to select, for each query term, how the query term should be used to score documents; which weighting schemes are best at indicating relevance for that query term. The results from this technique were generally better than the best combination of weighting schemes for the collections tested. This shows that selecting evidence for combination, through relevance information, can lead to successful combination of evidence.

Croft and Haines, [HC93], described RF in an alternative probabilistic model, the *inference network*. Inference networks are composed of nodes - representing documents, terms, phrases, etc. - and arcs representing the dependencies between the nodes. An example is given in Figure 15. The top nodes, labelled *d*, represent the documents in the collection. The nodes labelled *r* are concept recognition nodes, these nodes represent the content of the document. The nodes labelled *q* are query nodes, representing elements of the query. The bottom node, labelled *I*, is the 'information need' node. This

single leaf node corresponds to the user's information need; specifically it dictates how the query elements are to be used to score the documents (what operators are used in the query).

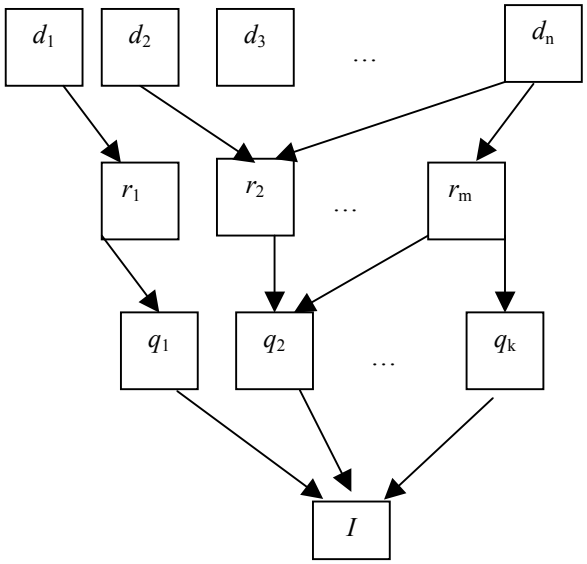

Figure 15: Inference network

Each node contains a 'link matrix' that calculates the belief for a node given the belief on its parent nodes. RF can alter the weights used to calculate the beliefs, in a manner analogous to term reweighting approaches. Query expansion is accomplished by adding new query terms as parents of the original query nodes.

Combination of evidence is possible in two ways: by using multiple representations in the concept recognition layer, e.g. single term and phrase versions of the same terms, and by the addition of query operators. An example of the latter is for the user to ask for documents containing the phrase '*information retrieval'* and any documents containing the word '*information'* and the word '*retrieval'*.

Haines and Croft's tested a number of RF variables: how to select terms for expansion, how to reweight terms, the relative weighting of query and new terms and number of terms to add. Although performance was variable across the collections tested, they found that query expansion and reweighting was effective. They also found support for Salton and Buckley's, [SB90], hypothesis that original query terms should be weighted higher than added query terms. They also provide limited support for the potential of RF in structured queries - ones that contain operators such as phrases and proximity information.

3.4.2 Multiple retrieval algorithms

Using more than one retrieval algorithm to score documents is a common way to combine evidential sources in IR. Simmonot, [Sim96], proposed a technique for selecting good retrieval algorithms techniques based on a user's relevance assessments. In her approach a number of indexing techniques were used to represent the content of documents, e.g. keyword representation, conceptual graph representation. Each indexing technique was associated with a retrieval algorithm. The user's query was submitted to each retrieval algorithm to obtain a number of document rankings and these rankings were combined to form a single document ranking that was presented to the user.

The user was asked to provide a set of relevance assessments, in a similar manner to standard RF. The degree of match between the rankings provided by the individual retrieval algorithms and the user's relevance assessments was used to score the *quality* of the retrieval algorithm for the search. This quality measure was then used to bias the combination in favour of 'good' individual retrieval algorithms. A low match between the user's assessments and an individual retrieval function's ranking resulted in that retrieval function having a low contribution to the combined ranking at the next iteration. A high match meant that the retrieval algorithm would give a high contribution to the

combination. The overall system then selected which retrieval algorithm was giving the best performance for the user at each feedback iteration.

Smeaton, [Sme98], suggests that retrieval strategies which are conceptually independent should work better in combination, and that retrieval strategies that work to same general level of effectiveness should be suitable for conjunction but again this is not always guaranteed to work. In particular, the results presented in [Sme98] indicated that conceptual independence of techniques in retrieving different documents did not appear to make a significant difference in experimental setting. However support for this claim is to be found in [RLVR02a].

3.4.3 Multiple feedback algorithms

For RF, a natural combination of evidence is to combine the results of different feedback methods. This could involve either combining the rankings given by different RF methods run on the same original query and relevance assessments, or combining the modified queries produced by several RF methods. This would be similar in spirit to Belkin et al.'s data fusion approach described in section 3.4. Lee, [Lee98], examined the former approach – combining rankings from multiple feedback functions, this will be discussed separately in section 3.5. in the discussion of relevance feedback without relevance information as this was the main area of Lee's work.

3.4.5 Summary of combination of evidence for RF

Combination of evidence has the potential to be a powerful technique for RF. However, the majority of techniques attempted have shown that combination of evidence is a very *variable* technique for initial retrieval. It will improve some queries but degrade the performance on others. In addition, it is also very difficult to predict what evidence to combine for different collections or queries. Using relevance information, section 3.4.1, to guide the combination process does seem to overcome at least some of these difficulties.

3.5 Relevance feedback without relevance information

RF, as described so far, depends on a user providing relevance assessments for a sample of the retrieved documents. An alternative approach, known either as *pseudo*, *blind* or *ad-hoc* RF, employs RF techniques to automatically improve a ranking before any documents have been shown to the user.

In this technique the system generates a document ranking from the initial query, selects a small number of documents from the top of the ranking, then initiates an iteration of RF by assuming these top-ranked documents are all relevant (the *pseudo-relevant* documents). The new query, generated by RF, is then used to produce a new document ranking which is shown to the user. The basis behind pseudo RF is that an iteration of feedback, based on the most similar documents to the user's initial query, will give a better initial ranking of documents.

This technique was first suggested by Croft and Harper, [CH79], as a means of estimating probabilities within the probabilistic model for an initial search[23]. It has since been widely investigated as a technique for improving document rankings. Croft and Harper also pointed to the fact that this method of improving a document ranking can suffer from one major flaw - *query drift*. Query drift occurs when the documents used for RF contain few or no relevant documents. In this case, RF will add terms to the query that are poor at detecting relevance, and hence in retrieving relevant documents.

The pseudo RF technique then, works well for 'good' initial queries - those that are good in retrieving relevant documents - and poorly for 'bad' initial queries - those that are bad at retrieving relevant documents. There are two possible solutions to this problem: either improve the initial ranking, so that there is a greater likelihood of relevant documents being used to modify the query, or improve the detection of relevant features, i.e. develop better RF techniques.

Mitra et al., [MSB98], have attempted, with some success, to rectify query drift by improving the precision at the top of the documents ranking, increasing the likelihood of actual relevant material being contained within the set of pseudo-relevant documents, and hence decreasing the likelihood of query drift. Their experiments used two approaches: a set of Boolean filters and term correlation information to prioritise retrieval of documents that covers all aspects of a query. They found that their approaches

[23] As a replacement for the *idf* term weighting function which is traditionally used when there is no relevance information.

work well for manually and automatically created filters, however around 25% of the queries still suffer from query drift.

Buckley et al., [BSA+95], also looked at improving precision at the top of the initial document ranking. They used massive query expansion (500 terms and ten phrases - commonly occurring pairs of words) from the top 30 retrieved documents. Their experiments produced significantly better results than with no feedback, particularly with respect to the precision of the new document ranking.

Most other researchers have concentrated on improving the feedback used in the pseudo RF approaches. Efthimiadis and Biron, [EB94], for example, found in their experiments that standard RF techniques used in pseudo RF experiments performed only slightly poorer than experiments using RF from users and with no feedback. However, the actual performance varied according to the algorithm used to rank terms for query expansion. Robertson et al., [RWJ+95], also found increased performance over no feedback, especially when using passages rather than the whole document, to select expansion terms

In [Lee98], Lee proposed an ad-hoc RF technique based on multiple RF techniques. The basic hypothesis is that, as different RF techniques may produce different modified queries, and different queries will retrieve different documents, then using a combination of RF techniques to modify queries will retrieve more of the relevant documents. An initial experiment was carried out treating the top 30 documents as relevant and using a vector-space retrieval function. This experiment compared the documents retrieved after performing pseudo retrieval using a Rocchio technique, Ide-dec-hi, F_4, a variant of F_4[24], and a simplified version of Fuhr's RPI formula, [FB91], Equation 17.

$$w'_{qi} = \log\left(\frac{p_i(1-q_i)}{q_i(1-p_i)}\right), p_i = \sum_{r=1}^{n_{rel}} \frac{w_{ri}}{n_{rel}}, q_i = \sum_{n=1}^{n_{nonrel}} \frac{w_{ri}}{n_{nonrel}}$$

Equation 17: Version of RPI used in [Lee98]

This experiment validated Lee's initial hypothesis: different RF techniques retrieved different documents although the different RF algorithms performed at approximately the same level of retrieval effectiveness. The similarity of the documents retrieved by each RF algorithm varied according to the RF technique used (e.g. the two F_4 techniques retrieved very similar documents but Rocchio compared with the modified F_4 formula only had about 50% of documents in common). The difference between the various RF techniques was also reflected in the query terms used to expand the query.

A second experiment combined the rankings, after normalisation of similarity values, obtained from the different modified query vectors. Combination of the rankings can provide significant improvements in effectiveness over the single RF methods. However more combination is not always better: combinations of two or three RF algorithms generally performed better than combinations of four or five RF algorithms. Given that the algorithms produce different rankings, after new retrieval, one might expect that the more different are the rankings, the better the combined performance. However, Lee's experiments did not generally demonstrate this conclusively.

Although the pseudo RF techniques described in this section can improve retrieval performance over not using pseudo RF, the problem still remains that it is a variable technique: some queries will be improved, others will be harmed. Several of the authors mentioned indicate that uncovering more details about the collection statistics, documents being used for RF and query characteristics may be used to predict which queries should be used for pseudo RF. For example, Lindquist et al., [LGF97] investigated various parameters for automatic RF using the vector-space model and found optimal performance was gained using between 5-20 documents and 1-20 terms for feedback. They also provide support for weighting new query terms against original query terms, using within-document term frequency and thresholding the query terms (only performing relevance feedback on queries that have terms with a high *idf* value). This leads to the suggestion that certain characteristics of a term may be good at predicting how the query is likely to improve given expansion by that term, which may be useful in pseudo feedback.

[24] [Rob86], Equation 12,

The queries that do well with pseudo feedback are those queries that are already retrieving relevant documents close to the top of a document ranking. However, those queries that do suffer from pseudo-relevance feedback are those that are already performing poorly; making these queries even worse may hinder the use of pseudo feedback as a standard retrieval technique. An alternative suggestion to pseudo feedback made by Buckley and Gay, [BG94], is to perform a high recall search and then a high precision search on the retrieved documents, thus trying to help poor queries before improving the order of retrieved documents.

4 Summary of automatic techniques for relevance feedback

In this section we summarise the work on automatic RF techniques. It is clear from the vast majority of work on automatic query modification that it can prove an effective, practical solution for improving the quality of on-line searching and it has been demonstrated to work well under a number of conditions. In particular, it is a very useful technique for improving the performance of short queries or queries which provide poor initial rankings. The basic approach of reweighting and expanding queries, using terms drawn from the relevant documents, works well with the major contribution often coming from the expansion component of the query modification [SB90], although this may be collection dependent.

Although there has been a large volume of theoretical work on RF, in the foundations to the probabilistic model for example, there remain a number of basic questions for which there are only heuristic solutions. For example, if we choose to add only a number of terms to the query, how should we choose how many terms to add? Similarly, how should we rank terms to give an optimal list of expansion terms? Functions such as F_4 that order terms by their discriminatory power are typically used for this purpose but the actual performance given by these functions, and by query expansion in general, is variable and is affected by collection, query and retrieval system used. Although the probabilistic model, section 2.2.3, gives a strong theoretical basis for ranking documents after relevance information has been provided, there is a lack of theoretical evidence to predict what makes a good set of expansion terms for a given collection-query-system combination.

One potential solution to this problem is to involve the user in the process of modifying the query. In section 1 we argued that one of the benefits of RF is that it requires minimal effort from the user - a user only has to *identify* relevant material not *describe* it. However we may gain a better representation of what material is likely to be relevant if we allow the user more control over the term selection process and also if we pay more attention to the tasks a user is trying to achieve with a system. These interactive aspects of RF are the topic of the next section.

5 Interactive query modification

All the methods for query modification described previously *automatically* extract terms from documents and add some or all of them to the query. A natural alternative is to allow *users* to select the terms to be added - *interactive query expansion* (IQE). The user, who has the best insight for determining relevance, then has more control over which terms are added to the query. The strength that is claimed for IQE is that the user can select better query expansion terms than the system. In this section we shall look at the basic research on IQE, section 5.1, examining how terms should be ranked for presentation to the user, section 5.2, and the effectiveness of IQE against automatic query expansion (AQE), section 5.3.

5.1 Fundamentals of IQE

In addition to investigation ranking functions for query expansion, Harman, [Har88], investigated the possible effectiveness of an interactive approach to query expansion. The experiments she carried out were designed to test how effective query expansion *could* be if the user selected expansion terms from a list of terms that were pre-selected by the system.

She performed an initial experiment, on the Cranfield 1400 test collection, in which a variable number of possible expansion terms[25] were added to the query. This experiment gave two main conclusions. First, she found that different methods of sorting the expansion terms gave different performance: some

[25]With no reweighting of the query terms.

methods for sorting terms were better than other methods. Second, and more importantly for IQE, the performance of query expansion varied according to how many terms were added to the query. For the Cranfield 1400 collection, expansion by 20 terms gave optimal effectiveness.

She performed a further experiment in which the system selected expansion terms from a list of those terms that occurred in at least one of the *unseen* relevant documents. This simulated a 'perfect' choice of expansion terms on behalf of the user - the system only added terms that would retrieve unseen relevant documents. This approach *(IQE-simulated*) was compared against the performance given by expansion using the top 20 expansion terms (*AQE*).

This IQE-simulated approach reduced the number of expansion terms from the 20 that were added in the AQE version to an average of 12 terms per query. Comparing AQE and IQE-simulated, Harman found that, although the AQE worked well and gave large overall improvements in retrieval effectiveness, the IQE-simulated expansion was capable of improving these results further. In addition, the IQE-simulated expansion was more consistent in improving performance. This latter finding was important: automatic query expansion (AQE) shows good overall performance when averaged over a set of queries but this performance increase is variable, some queries do very well with AQE others improve very little or suffer a degradation in performance. IQE as Harman deployed it, on the other hand, improves more of the queries.

Harman explored alternatives for obtaining terms for query expansion: query expansion by term variants, expansion by nearest neighbours. The first method - expanding the query by all variants of the query terms - showed little improvement when performed automatically, i.e. adding all variants of query terms. However using the 'perfect user' strategy Harman did obtain significant improvements. The second strategy - expansion by similar terms as given by co-occurrence information - also showed a drop in performance when performed automatically but an increase when performed in the simulation of a perfect user. Harman also demonstrated that *combining* query expansion techniques can further improve performance.

Harman's 1988 experiments only examined query expansion: the expansion terms were not weighted according to their utility in retrieving relevant documents. In [Har92b] she ran a series of experiments on the same collection as in [Har88], the Cranfield 1400 collection, to determine the relative effectiveness of expansion and reweighting. She showed that, on this collection at least, expanding the query is more important than only reweighting query terms. Combining both techniques will give best overall performance. The relative merits of term reweighting and expansion may differ between collections and models but probably generally hold. She also demonstrated that multiple iterations of RF can increase performance over single iterations, so RF is useful over the course of a search.

The work on AQE demonstrated that, although RF can improve retrieval effectiveness, it is variable across queries: some queries do very well with relevance feedback whereas others can show degraded performance. In IQE it might be reasonable to assume that a user can negate this variability by selecting only good RF terms and ignoring the non-relevant ones. This potential benefit raises a number of questions regarding how good AQE methods are for IQE purposes. In the following sections we shall examine how ranking terms for IQE can affect performance, and the relative effectiveness of AQE and IQE.

5.2 Ranking expansion terms in IQE

It may be that the traditional term ranking algorithms used for AQE will perform differently when used by real subjects. That is, techniques that are successful in automatically selecting expansion terms are not suitable as a basis for a user selecting terms. One reason for this is that the reasons for a user selecting a term may not be based only on retrieval effectiveness. A user may, for example, choose fewer expansion terms due to the increased effort of term selection, or may choose terms that refine rather than modify a search topic.

Efthimiadis, [Efth93, Efth95], examined eight term ranking algorithms, and investigated their performance in an IQE environment, when users performing real searches were making the relevance assessments and term selection. Four of these algorithms (F_4, F_4.modified[26], $w_i(p_i - q_i)$[27], and

[26] F_4.modified is the version of the F_4 weighting function that adds 0.5 to each cell in the numerator and denominator to prevent 0 entries (section 2.2.3)

EMIM[28]) have already been discussed. The fifth - Porter's algorithm, [PG88], - is similar to the F_1 function – section 2.2.3, placing emphasis on frequently occurring terms in the relevant set. This is shown in Equation 18.

$$Porter_i = {}^{r_i}\!/_{R} - {}^{n_i}\!/_{N}$$

Equation 18: Porter term weighting function

where r_i = number of relevant documents containing term i
R = number of relevant documents
n_i = number of documents containing term i
N= number of documents in the collection

The sixth algorithm - the ZOOM frequency measure [Mar82] - ranks terms by their total frequency of occurrence in the retrieved set. All within document occurrences are also included so this measure ranks terms by the total frequency within a set of documents. Ties between equally frequent terms are resolved by ranking terms alphabetically.

The seventh algorithm, *r-lohi*, ranks terms according to their frequency of occurrence in the relevant set of documents, resolving ties by the *tf* value of the terms (low *tf* to high *tf*). The final algorithm, *r-hilo*, is identical to r-lohi except that it resolves ties by ranking from high *tf* to low *tf* value.

In the data collection section of these experiments, Efthimiadis's subjects were asked to mark all potentially useful expansion terms and the five best terms. The terms were selected from documents that the user had assessed as relevant during relevance feedback. Efthimiadis evaluated the performance of the eight term ranking algorithms by comparing the rankings given for each query against the list generated by the users. For this, he used three criteria.

i. *comparing systems and user's ranking of term utility.* The first test looked at *where* the user-selected terms appeared in the system's ranking of terms (the top 25 terms give by EMIM, Porter, etc). Term ranking algorithms that have more user-selected terms further up the ranking are better than those algorithms that place user-selected terms further down the ranking of terms.

The most finely-grained test split the system generated list of terms into three sections (top, middle, bottom). The user-selected terms showed a distribution of 20%-30%-50% (20% of terms in bottom third of system ranking, 30% in middle third, 50% in top third) for all measures except ZOOM (with a distribution of 30%-30%-40%) and r-hilo(40%-30%-30%). The *wpq*, EMIM and r-lohi performed at very similar levels, followed by Porter, and, slightly behind, the two F_4 variants. The same analysis was performed for the five best terms identified by the users, which showed similar results: *wpq*, EMIM and r-lohi performing best, followed by Porter, then the F_4 variants, and finally ZOOM and r-hilo.

ii. *examining top five ranked terms.* The second analysis examined the top five terms in each ranking to compare the *similarity* of the term rankings. The result showed that pairs of algorithms (*wpq* and EMIM, F_4 and F_4.modified, Porter and ZOOM) were very similar. The terms of r-lohi are similar to *wpq* and EMIM, whilst those of r-hilo are more close to those of ZOOM than anything else. In certain cases, e.g. *wpq* and EMIM, the top five terms are almost identical with only the ranking differing slightly. The major differences were between the F4 cases (mostly influenced by *n*) and the other algorithms (mostly influenced by *r* and only different is when *r* is tied).

iii. *mean of their rank position of user's five best terms.* The rank position of the users' five best terms were summed to determine which algorithms gave the best ranking of these important terms. The results (*wpq*, EMIM > r-lohi, Porter > F_4.modified >F_4 > ZOOM > r-hilo) also highlight differences between pairs of algorithms but there were no significant differences between the superior *wpq*, EMIM, r-lohi and Porter algorithms.

Each of these analyses were designed to test how good the algorithm was at ranking terms for IQE. In each case *wpq*, and EMIM performed best with Porter and the F_4 variants performing well. The ZOOM

---

[27] Abbreviated, for convenience, to *wpq*, section 2.2.3.

[28] Section 3.1.

and r-hilo measures scored lowest in all cases. These results substantiate the relative merit of the algorithms derived for AQE when used for IQE (*wpq* and F_4). They also highlight Robertson's original concern that functions designed to measure discriminatory power of existing terms (F_4) were not necessarily the best to use in selecting new terms, as shown by the better performance of *wpq* over F_4.

5.3 Performance of IQE against AQE

Harman's original proposal for IQE was that user selection of expansion terms could give better performance than automatic expansion by the system. This may be true for a number of reasons. For example the system will typically base its estimate of term utility on very little relevance information which could lead to a poor set of expansion terms. A user, on the other hand, will be better able to filter out poor terms and only use those s/he feels are appropriate.

Harman, [Har88], demonstrated that selecting terms could improve retrieval effectiveness in a *simulated* case. Magennis and Van Rijsbergen, [MVR97], extended this study in two ways: by studying the *degree* to which IQE can theoretically improve performance over AQE and whether this theoretical improvement can be realised with actual users.

Magennis and Van Rijsbergen's experiments to determine the theoretical performance of IQE are based on Harman's [Har88] notion of a perfect user choice. The choice of a different test collection (the larger Wall Street Journal (WSJ) collection) necessitated repeating some of Harman's work. In particular they investigated how many terms to add[29]. They found that the range of terms, to *automatically* add to the query, to achieve optimal performance is closer to 0-10 for the WSJ than Harman's 20-40 terms for the Cranfield 1400. This shows the difficulty of predicting good estimates of numbers of expansion terms, in particular for different collections and different query sets.

Magennis and Van Rijsbergen repeated Harman's simulation experiment, which expanded the query using terms chosen from the relevant documents in the top 20 retrieved documents. They ranked the top 20 terms chosen from the relevant documents, and added the top *n* terms. Terms were weighted according to their presence in the unseen, or *target*, relevant documents as the function of query expansion is to select terms that are good at retrieving these new relevant documents. The cut-off value, *n*, was treated as an experimental variable with five values: 0 (no expansion) 3, 6, 10, and 20 (no selection of expansion terms). For all queries, each *combination* of cut-offs was tried. AQE systems will generally expand every query by the same number of expansion terms. As a user may expand each query by a different number of expansion terms, combinations of cut-offs were used to establish the best cut-off for each query. For example, expand query one by 0 terms, expand query two by 10 terms, query three by six terms, etc. Combinations, therefore, allow the simulation of a user adding a variable number of expansion terms. The experiment was run over four iterations of feedback and the best retrieval effectiveness was taken as the performance that could be expected by an experienced user.

The best retrieval effectiveness (precision over 100 documents retrieved) for the AQE case was achieved by adding the top 6 expansion terms. This method improved precision over automatic expansion by all 20 terms. The experienced user simulation outperformed both automatic expansion by the top 6 and by the top 20 terms. Moreover, the simulated experienced user selections improved the retrieval effectiveness for more queries: it was a more *stable* improvement over the AQE methods.

The experiment also compared the performance of the experienced user against Harman's original proposal, [Har88], of adding any term that appeared in a relevant, unseen, document. Harman's technique worked well against expansion by the top 20 terms, but only marginally better than automatic expansion by the top 6 terms, and less well than Magennis and Van Rijsbergen's approach. This supports Harman's 1992 conclusion, [Har92b], that term weighting (as was done in [MVR97] but not [Har88]) is important for query expansion.

A second experiment was run, using the same queries and same test collection, in which experimental subjects were asked to select expansion terms. This was designed to test the actual performance of IQE when relatively inexperienced users were making the term selection decisions. The subjects could add up to 20 terms, (the default being no expansion) and were allowed four iterations of RF. The searchers

[29] Using the F_4 measure to rank terms.

were asked to assess relevance but the test collection relevance assessments[30] were used to generate expansion terms. This was to ensure that the terms used for expansion were the same for all users, and were the same as in the experienced user simulation. This aspect of the experiment was hidden from the searchers.

For all queries, the users failed to reach the potential effectiveness of the simulated user and on the whole failed even to reach the level of AQE. So although IQE *can* improve retrieval effectiveness and *can* demonstrate consistent improvement over a set of queries, the subjects in this set of experiments failed to demonstrate the ability to make good term selections. This is a vital point for IR: if IQE is to realise the experimental potential demonstrated in Harman's earlier experiments, it is necessary to facilitate the selection of good query terms. How this process of iteratively developing a query can be made easier requires a more careful analysis of what processes users follow within IQE. We look at this in the next section.

5.4 Using IQE

In this section we present three investigations on user behaviour when interacting with an IQE system. The results from these investigations are not consistent. However the very lack of consistency across the experiments highlight important aspects of IQE and user interaction. They also highlight the fact that it is difficult to predict, or make assumptions, about what functionality users want from IQE or IR systems.

Beaulieu, [Beau97], as part of the ongoing work on the Okapi probabilistic system, carried out an investigation of three interfaces to IR systems. One of these only offered AQE, two offered IQE. The systems, unlike many query expansion systems, were not investigated through laboratory investigation but through operational investigation: the systems were used as an interface to a university library catalogue.

The first interface offered only AQE. The user was asked, for each document viewed, if the viewed document was similar to what documents s/he would like to retrieve. If the user's answer was yes, then they were offered the option of searching for similar documents. The query modification was hidden from the user; the users only saw the results of the new search. In operational trials, the uptake rate was around 33% percent (number of users trying the AQE option) and this led to retrieval of further relevant items in around 50% of the searches[31].

The first IQE system was based on a series of overlapping windows with separate windows for query, relevant titles, and the retrieved set of titles. The user was asked the same relevance question as in the AQE case ("Is this the sort of thing you are looking for? Y/N"). If the user answered yes, the document title was added to a list of titles of relevant documents. Users requested term suggestions by the use of an Expand Search button that caused the system to extract the top 20 expansion terms for display to the user. Users could then select those terms that they would like to use in a modified query. Uptake on this system was only 11% and query expansion only led to the retrieval of further relevant documents in 31% of the searches in which users tried IQE.

The results are significant for a number of reasons, relating to both the performance and behaviour of the IQE system. The take-up rate (number of users using query expansion) and the increase in relevant documents found after query expansion were both lower in the IQE system than with AQE. Users tended to select terms very strictly, with 50% of users reporting that they found it difficult to select appropriate terms, and around 25% of users editing their original query rather than modifying their query through the IQE facility.

A third interface was developed to give the user more information on which to base their choice of term selection. A number of changes were made to the system design:

i. the overlapping windows design was replaced by a multiple pane single window design.
ii. an interactive thesaurus component was added which allowed the users to view terms related to the initial query terms.

---

[30] These were the relevance assessments associated with the WSJ test collection, rather than the assessments given by the users in the course of the experiment.

[31] Measured by analysis of search logs.

iii. a separate working space was included to view the developing query. The source of query terms was also colour coded (initial query, IQE added query, user added query, etc.)

iv. each time the user made a relevant document selection the interface was dynamically updated to show the effect of choosing this document.

The premise behind this interface was that the user would gain more information on the effects of actions such as making relevance assessments. The uptake rate for this system was 19.5% and it led to the retrieval of further relevant items in 46% of the searches. This system had higher take-up and effectiveness rates than the first IQE interface but the figures are still lower than the AQE interface. The indication is that, although an improved interface can increase the level of use of IQE and the effectiveness of term selection, it remains an open problem how to get users to employ IQE in operational environments.

Beaulieu and Jones [BJ98] extended this study by looking in more detail at three factors that affect interaction: functional visibility, cognitive load and balance of control between the user and system, specifically relating them to this set of experiments. The functional visibility - allowing the user more information on how the system works - is important at two levels. Not only must the user be aware of what options are available at any stage but they must also be aware of the *effect* of these options. For example, the initial IQE interface was more difficult for user as it separated the act of modifying the query and that of assessing relevance.

The cognitive load, or effort that a user must put into an action, may deter the user from trying an action that would be beneficial such as choosing more query terms. Cognitive load is also related to the notion of *control*: generally the more control the user has the higher the overall cognitive load is placed upon the user. Thus, as Bates [Bat90] reported, the balance of control, between the system and a user, is a question not necessarily of how much control the user has but over what to give the user control. In this context it may be preferable to use AQE as a default expansion technique, and to use IQE as an option for certain types of search or search stage, rather than use a single method of query expansion.

Fowkes and Beaulieu, [FB00], in a separate investigation, hypothesised that the complexity of the search may be an indicator of when to use AQE or IQE. Searches for which the desired information is clearly defined and for which the user can retrieve relevant information easily benefit more from AQE. Searches for vague information needs or in cases where little relevant information is being retrieved benefit more from IQE. In addition, users are more likely to employ IQE in a complex or difficult search. A related point is that users may employ RF, either AQE or IQE, less often when the retrieval system is performing well – when it is easy to retrieve relevant information.

Belkin and Koenneman [KB96] also investigated the use of IQE versus AQE. In this study they looked at the performance and behaviour of 64 novice users in the use of three different types of RF mechanism: completely automatic query expansion, automatic which showed the expanded query after retrieval, and interactive which allowed users to modify query before re-evaluation. They also had a no-feedback control and each user was trained on this baseline system. On the whole the findings were positive: the subjects who could control the expansion terms (the third, interactive, case) had better performance, and feedback itself gave better performance than no feedback. Users tended to choose semantically related feedback terms, and entered fewer terms manually than were suggested automatically.

This set of experiments demonstrated that interactive expansion could give positive results over automatic expansion. One particular feature of the experimental design may hold the key to the experiments' success. The task that users were given was to develop a good query for an information filtering system[32], 'good' in this sense meaning one which was good at retrieving relevant documents. The task the users were given, then, was one that concentrated the users' attention on the development of good queries, a situation that would lend itself to the use of techniques such as IQE. How to encourage users to develop good queries and develop more sophisticated queries does remain a difficult area as shown by Beaulieu et al.'s experiments.

Dennis et al, [DMB98], in a study looking at different types of query expansion techniques found that although users could successfully use novel expansion techniques and could be convinced of the

[32] An information filtering system matches a query (or search profile) against a changing set of documents. Most IR systems operate on a fixed set of documents.

benefits of these techniques in a laboratory or training environment, they often stopped using these techniques in operational environments. The question may be, then, can we design systems that will lead users into spending time developing queries through IQE.

5.5 Summary of interactive query expansion

In this section we summarise the case for IQE over AQE. The general intuition that some increased control for the user in selecting query expansion terms would be beneficial seems to be valid. Although systems have access to internal statistical information that allows them to select good discriminatory terms, users can make more informed *relevance* decision. The question is how this process of query modification should be constructed to translate the potential benefits of IQE into actual increases in retrieval performance.

There are several issues involved in this problem. The first is to decide what is the actual role of the user: should we ask the user to interactively create queries or perform an editing role on system-generated queries? How much of the query-generating process should be interactive and at what stages should we expect and desire user involvement?

Several of the reasons given by users for not using AQE are also applicable to IQE, [BCK+96, RTJ01], e.g. these are time-consuming actions, the relation between cause and effect is not clear and on what principles the selection of terms should be made is not obvious. The latter point – how terms should be chosen – is significant. It may be the case that users are better at eliminating potentially poor terms than they are at selecting good terms for query expansion. IR systems need to be able to help users make difficult decisions regarding term quality.

In the next section we shall describe interfaces that were specifically designed for RF. These interfaces are an attempt to overcome the user's reluctance to initiate RF. The success of interactive approaches to RF may, of course, not simply be a result of the interface or algorithms used by the system. For example the characteristics of the user, such as experience with on-line searching, and the search itself may affect the use and the success of more user-oriented methods of interaction. We shall examine some of these characteristics in section 7.

6 Interfaces and RF

The reluctance of users to engage in RF often comes from a poor understanding of why RF may be useful and how RF should be used in a search. This may be because RF is presented as a separate task to querying and to assessing retrieved documents. In the next two sub-sections we discuss two systems that attempt to incorporate RF as a seamless task – the process of RF is integrated into querying and assessment of documents.

The two approaches have a common underlying principle: each relevance assessment given by the user initiates a cycle of RF. The major difference between the two approaches – incremental feedback, section 6.1 and ostensive browsing, section 6.2 – is the interface design and principles.

6.1 Incremental feedback

Most RF systems treat the process of relevance assessment as a batch process: users are shown a set of documents and provide relevance assessments on a number of documents before requesting RF. Aalsberg, [Aal92], proposed the alternative technique of *incremental* RF. Rather than asking a user to batch process relevance assessments by assessing a *number* of documents in a ranking, he suggests presenting only one document at a time. The user is asked to make an assessment on the displayed document before being shown the next document. With each relevance assessment made by the user, the query can be iteratively modified through feedback. The formula used by Aalsberg simplifies the Rocchio, Ide-dec-hi and Ide-regular formulae[33] to the one shown in Equation 19.

[33] Section 2.2.2.

$$Q_{i+1} = \begin{cases} \alpha.Q_i + \beta.D_j & if\ rel(D_j) \\ \alpha.Q_i - \gamma.D_j & if \neg rel(D_j) \end{cases}$$

Equation 19: Iterative RF

where Q_i = query for iteration i, $Q_i + 1$ = query for iteration $i + 1$,

α and γ are weights to bias retrieval in favour of the query or relevance information

This technique does not require the user to explicitly request RF, thus side-stepping the difficulty of getting users to interact. However it may not allow users to make *relative* relevance assessments, which has been shown to affect users assessments and method of making relevance assessments, e.g. [FM95, EB88]. The particular implementation also forced users to make a relevance decision. Users, however, may not always be able to decide on the relevance of a document at the time they view it.

The model was tested in [Aal92] against Rocchio's formula, the Ide-dec-hi and Ide-regular. The model was also tested against Ide's *variable* RF, section 2.2.2. This model forms a new query from the first relevant document and all preceding non-relevant documents. This is, then, analogous to the Ide-dec-hi that uses all relevant and the first, retrieved, non-relevant document, section 2.2.2. The test collection evaluation showed iterative RF can perform better than the Rocchio, and Ide-variants but performs roughly the same as variable RF.

In a separate experimental investigation Iwayama, [Iwa00], suggests that incremental relevance feedback of the form proposed by Aalsberg works better for well-specified topics. These are topics for which the set of relevant documents has a high similarity. This is because iterative feedback retrieves documents that are very similar to the ones used for feedback. It does not, however, perform as well in retrieving relevant documents that cover a number of topics.

6.2 Ostensive browsing

Campbell's ostensive weighting technique, described in section 3.2, was combined in [Cam99] with a novel browsing interface, an example of which is shown in Figure 16.

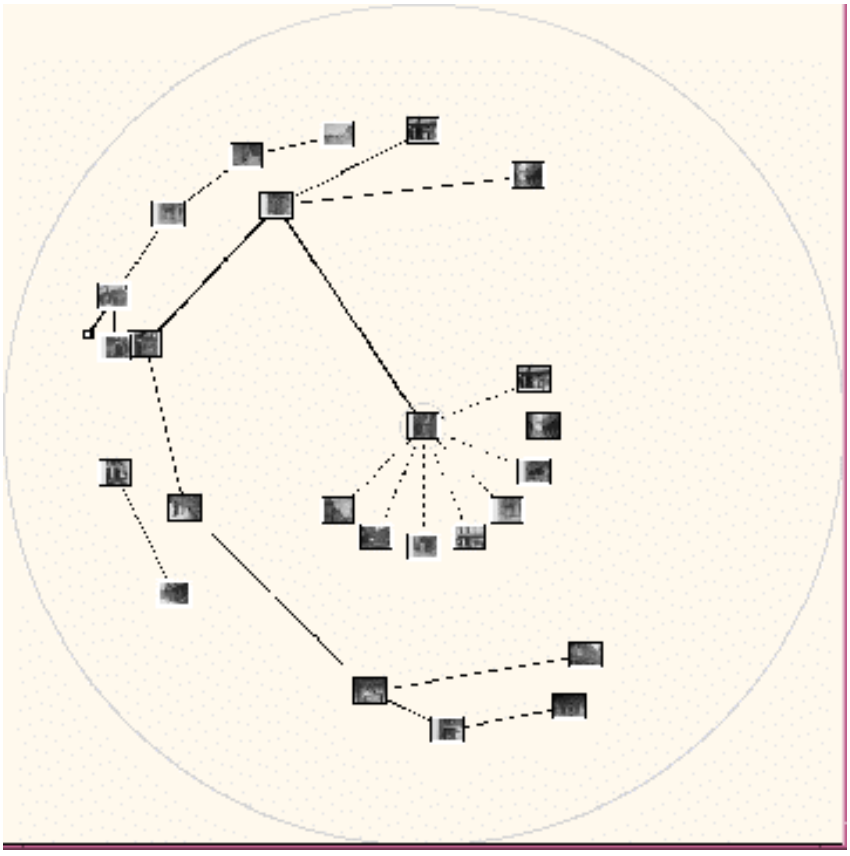

Figure 16: Ostensive browser interface, taken from [Cam99]

This interface contains two features: paths and nodes. A node consists of a retrieved object. In Figure 16 these objects are images. Clicking on a node will cause the system to perform a RF iteration using all the objects in the path that contains the node. A small number of the top retrieved objects are then displayed to the user, who may choose to continue the path by clicking a new object or return to a previously followed path. If a user selects more than one retrieved object, this corresponds to a diverging path: two paths with the same initial components.

Each selection of a node by a user is taken to be an implicit relevance assessment or expression of interest in the object by the user. No explicit request for RF is necessary by the user. The paths themselves correspond to multiple iterations of feedback; each object is the result of RF performed on the objects preceding it in the path. Objects may appear in different paths as the result of being retrieved in response to different RF-modified queries.

This is similar to an extent to the iterative method of RF described in the previous section in that only one additional document is added to the relevant set at each iteration. The major interface difference is that the user is not asked to make an explicit assessment of relevance or decision on the relevance of a document. The major implementational difference is that Campbell uses the ostensive weighting extension to the probabilistic model, described in section 3.2. The use of paths also means that RF decisions are reversible: the user can backtrack to a previously selected document at any point in the search.

One of the main aims of Campbell's work on ostension is to remove the need for a user to manipulate a query. However this also removes the *control* from the user in modifying the content of the query. A user cannot manually manipulate the query as is generally possible with the traditional RF systems. Whether or not this hiding of the IR system's functionality benefits the user or not requires further investigation.

Both the interfaces described in this section force users to employ RF. However, in most interfaces the user has RF as an option. As shown previously users can be reluctant to initiate relevance feedback iterations. Partly this is because the decisions made by RF are not clear to the users and the possible effects of RF are not obvious before initiating feedback. Ruthven et al. [Rut02, RLVR03] developed an interface to an RF system that used explanations to help users understand what decisions RF had made and why these decisions had been made. An example of an explanation is shown in Figure 17. The results from these experiments indicate that presenting a more meaningful description of RF can lead to more use of feedback techniques by the searcher.

> *'As you have not found many useful documents, I have added the following words to try to broaden your search* ***couldst inescapeably hillle banquo macduff laurenson****. You can remove any word you do not think is useful for your search'.*

Figure 17: Explanation of RF

Much more experimentation is required into good interfaces for RF; ones that encourage users to initiate feedback and make good relevance decisions. In particular this need for further experimentation is necessary because the range of factors that lead to the success or failure of interaction with an IR system are very diverse. Many researchers have argued that the process of retrieving relevant information is richer and more complex than the relatively simple model described so far, e.g. [Bat90, Kuh93, Ing92]. In the next section we shall outline some of the features of user searching characteristics that affect how RF is used and its success in improving searching.

7 User issues

We can separate out some factors that will affect success of failure of RF algorithms: user experience of on-line searching, section 7.1, user characteristics, 7.2, and the process of making relevance assessments and term selection, section 7.3.

7.1 User experience

In [CPB+96, KQC+95] Cool, Koenemann et al looked at the effects on new types of IR systems (ranked-output, best-match model) on the searching behaviour of users who were expert in Boolean,

exact-match searchers. Their aim was to examine how searchers who were experienced on one type of searching system performed when they searched on a different search system.

Ten searchers were each given five different TREC topics and asked to provide a filtering query for each topic. They were analysed both on performance - how well they did in providing an effective query - but also on how they utilised new features such as automatic RF and special operators (such as synonym operators) provided by Inquery, [CCH92], the underlying retrieval system.

The results showed that users *can* make use of the new features but that the take-up rate was variable. Some searchers used the new features from the beginning, and used them effectively. Other searchers learned to use the features, what Cool et al classified as 'combining old search strategies with new ones'. Some searchers only attempted to use their existing search strategies in the new environment. The indications from this experiment is that the best-match systems, which offer relatively weak indications of how they should be used[34], may need to offer users more explanation on how they operate. This latter conclusion was also demonstrated in [WRJ02], who indicated that users may be unaware of basic best-match principles, such as ranking of documents, when interacting with IR systems. Additional general conclusions from [CPB+96] were that interactive searching seemed to provide worse results and that users may need a mental model of how the system works to use it effectively. However, as reported by Belkin, [Bel97], even if people understand RF conceptually they can have difficulty in controlling it in operational systems.

Experience on individual IR systems is important, general experience with any kind of IR system is also important. Hsieh-Yee, [HY93], investigated the effect of subject searching experience and topic familiarity in interactive searching with particular reference to how searchers selected search terms. Her results indicate that experienced users - those with more than one year's searching experience, or who have attended a course in on-line searching – differed in two ways from novice users.

Firstly, experienced users were more flexible in their search strategies than novice users. Measuring the strategies used by the searches, using Bates's [Bat90] categorisation of search tactics, section 7.1, Hsieh-Yee noted that novice users were more consistent in their search strategies whether the search topic was familiar or unfamiliar. However, experienced searchers were more likely to use different strategies according to how familiar they were with the search topic. Secondly, experienced and novice searchers selected terms differently. Experiences searchers used more synonyms and concentration on combining search terms than novice users. When searching on an unfamiliar topic novice users depended more on their own search terms, whereas experienced searchers used tended to use more thesarual terms, prepared term selection more heavily and spent more time preparing a search.

The major conclusion for IR from this study is that the user's experience level has a strong affect on how a user searches. A particular conclusion for RF is that novice and experienced users may require different methods of selecting expansion terms or may require query expansion to be described in a different manner. We shall return to factors that affect a user's selection of terms in section 7.3.

7.2 User characteristics

We should also consider the characteristics of searchers. Borgman, [Borg89], reviewed a range of user characteristics that may play a role in determining the success or failure of online searching in IR systems. Borgman's analysis concentrated on Boolean searching but a number of the aspects she examined such as technical aptitude, educational background, personality type and the retrieval task will be pertinent to all interactive searching.

Other individual aspects that affect searching behaviour include the task the user is trying to achieve [VH00], and the searchers professional discipline [Fid91]. Heuer, [Heu99], also suggests that people in different domains use information differently. In addition, Heuer suggests that people often want better quality information rather than simply more information so adequate techniques to cut down the amount of potentially relevant information may be important.

Peters, [Pet89], examined the transaction logs of an OPAC library system to classify searches in which no results were obtained. A high proportion of the searches (39%) were due to documents not being in

[34] Although the Boolean model may be more difficult to use, the fact that it forces the user to structure their queries may actually make it easier to understand how to interact with the system.

the database, a further 20.8% were due to typographical errors or spelling mistakes[35]. The majority of other errors were due to problems with the system such as author searches for titles, and misuse of controlled vocabulary. The error rates were high, as high as 46% of searches, so this is relatively severe, even though the systems themselves were popular. Common problems included low use of advanced features or poor understanding of how to use the systems.

Part of this difficulty in using IR systems is that different types of knowledge are required for different tasks. Borgman, [Borg96], for example, identified three types of knowledge necessary in information seeking:

i. *conceptual* knowledge of the information retrieval process. This is knowledge necessary to translate an information need into a searchable query.

ii. *semantic* knowledge of how to implement a query in a given system. Once the user has established what concepts and terms are to be used to form a query these elements must be converted into an appropriate query for the system. This requires knowledge of how and when to use the system features.

iii. *technical* knowledge. This covers basic computing skills and the knowledge of the query language.

A user's lack of knowledge may not only hinder search effectiveness but may also require the user to interact ineffectively with the system. This problem also relates to the earlier discussion on interactive query expansion: the presentation of what the system is trying to achieve is important for effective interaction with the system.

7.3 Feedback, term selection and relevance assessments

The success of RF depends largely on two components: the user's evidence as to what constitutes relevant material and the quality of the RF algorithm. In this paper we have concentrated mainly on the latter component – the RF algorithms themselves. We have briefly discussed some of the factors, such as the interface, which can have an affect on the former component – the relevance information given by the user.

The information given by the user is vitally important in helping the RF algorithm make good query modification decisions. In this section we shall outline some of the studies that have examined how users give relevance information. In particular we shall concentrate on what types of feedback users employ, section 7.3.1, how user's choose query terms, including terms chosen during feedback, section 7.3.2, and the factors that affect a user's relevance assessments, section 7.3.3. These sections are intended to highlight aspects of the users' interaction that can affect the quality of information given to a RF system.

7.3.1 Types of feedback

Spink [Spi97] looked at the various types of feedback in mediated[36] Boolean information-seeking sessions. Based on her study of 40 searches, she proposed a classification of five types of feedback. These are not all types of *relevance* feedback; they also include query modification actions that are intended to modify the search in some other way. Her classification of feedback types is:

i. *content* relevance feedback. In all the searches studied the user and intermediary used the content of documents to make relevance judgements. The judgements could be either negative or positive. This is the second most common type of feedback and was the only type of feedback where the users' judgements were more common than the intermediaries' judgements. Based on content relevance feedback searchers could modify their query and re-search.

ii. *term* relevance feedback. This was the identification of new search terms by the user or intermediary from the relevant material. This is equivalent to the common notion of RF discussed in

[35]This may be a particular problem for Boolean systems in which one mispelt query term can result in an empty result set.

[36] Mediated searches are those in which a professional searcher, such as a librarian, aids a searcher in formulating queries.

this paper except that the new query terms are selected manually. This was used fairly evenly across searches, i.e. intermediaries and users employed it in approximately the same number of searches but intermediaries tended to use the technique more often within a search. This type of feedback was used far less often than content or magnitude feedback.

iii. *magnitude* feedback. Magnitude feedback refers to feedback based on the size of the retrieved set of documents. Judgements were that the retrieved set was too large, too small or just about right. This type of feedback was the most common observed feedback type. Intermediaries used this type of feedback in all searches; users initiated magnitude feedback in around three-quarters of the searches. However the intermediary made around 81% of all the magnitude feedback decisions. Thus, it appears that the intermediaries were more concerned with the size of the retrieved set than the users who were more interested in the relevance of the documents, a point also noted by Shenouda, [She91]. This kind of feedback is not exclusive to Boolean systems, e.g. White et al [WJR02] reports similar findings on best-match systems in a small study of searches on Web search engines.

iv. *tactical* review feedback. This type of feedback (6% of total feedback instances) was based around search strategies. Specifically these involved the intermediary examining the search history to make a decision about how to proceed with the search, e.g. to avoid repeating a previous search. This may not be an operation that is likely to be performed by an inexperienced user of the system.

v. *terminology* review. This type of feedback, corresponding to around 1% of feedback instances, involved the intermediary or user making a strategic decision by looking at terms in inverted file. For example the intermediary may search for alternative spellings of query terms.

The importance of studies such as this is that they indicate that users often want to give information on more than just the content of the documents: *relevance* feedback is not the only important feedback but is often the only feedback that is considered or offered by the system.

7.3.2 Sources of query terms

Relevance feedback is not the only source of query terms after a user has performed an initial retrieval. The user may add more terms or may select terms from other sources, such as a dictionary. The relative effectiveness of feedback terms against terms from other sources has been addressed in a study by Spink and Saracevic [SS93, SS97].

Spink and Saracevic investigated the use and effectiveness of search terms gathered from various sources (query, user interaction, term RF, thesaurus, intermediary) during 40 online mediated Boolean searches. The search logs were analysed for the first occurrence of each query term, this was taken to be the source occurrence of the term for the purposes of classifying the source of the term. Repeated uses of the same term were ignored.

The users were responsible for most (62%) of the search terms but only 38% of the terms came from the user's initial query statements. That is the majority of search terms came from the interaction with the IR system (after the users have written their information needs). 19% of search terms came from the thesaurus, and only 11% of search terms came from term relevance feedback. This is a rather low percentage of terms coming from the relevant documents, particularly as the intermediary-selected terms comprised the majority (65%) of terms chosen using feedback. Term relevance feedback was not automatic in this study: terms were chosen manually from the relevant items, which may cause the low reported percentage of use. Relevance feedback was only used in about half the searches; again, this would seem to be a low percentage.

The single most successful source of query terms for retrieving relevant documents was the users' query statement. Terms from this source retrieved half of the relevant items. Term relevance feedback was poor at retrieving relevant items, either on its own or in combination with other sources of terms. However, RF, unlike other sources of query terms such as the thesaurus, was more likely to improve rather than degrade a search's performance. RF appeared to be used in certain circumstances, for example it was often used later in search when there was more interaction or when the user had exhausted the other search options.

This study contains some relatively negative findings for RF, especially the lack of early up-take of RF. This may be tempered slightly by the fact that the users did not have the opportunity to explore automatic RF, which might have facilitated more interaction with feedback terms.

7.3.3 Relevance assessments

The final aspect of information-seeking we shall address, although briefly, is the process of making relevance assessments. RF algorithms require users to assess a sample of the retrieved documents but the criteria under which a user makes a relevance assessment can be subject to a number of factors. In this section, we shall introduce some of these factors.

One of the main factors is the *order* in which documents are shown to the user. Several studies, e.g. [FM95, EB88], point to the importance of the position of a document in a ranking when assessing the relevance of the document. Relevance assessments are relative: viewing one relevant document can change the user's perception of the relevance of subsequently viewed documents. Tiamiyu and Ajiferuke, [TA88], also looked at the effect that the order in which relevance assessments are made can have on retrieval performance. They suggest three types of dependence that can exist in retrieval;

i. *independence*. Each document should be considered as an independent relevance assessments,

ii. *complementarity* relationship. The information contained within two documents sums to more than the sum of relevance ratings of each document together.

iii. *substitutability* relationship. The information in one document can substitute for the information in another document.

They show, theoretically, that the presence of different types of relationships can, although, giving same recall-precision results, give a very different result for user satisfaction. This also brings up the question of whether we should treat all relevance assessments as a single set of assessments. Draper, [Dra00], for example makes the point that users typically assess *individual* documents as relevant, not a group of documents, whereas RF systems treat relevant documents as a set of related items.

Janes, [JJ91], also demonstrates that different *representations* of documents (title, abstract, full-text) can affect relevance assessments, meaning how the document is presented can affect how likely it is to be assessed relevant.

Relevance assessments are often treated as *binary* assessments: a document is either relevant or not relevant. However, in practice, documents may be regarded as more or less relevant than each other: relevance assessments are often *partial* assessments[37]. Spink et al, [SGB98], examined relevance assessments from four separate studies of information seeking to examine the role of partial relevance assessments. In particular they looked at whether the use of partial relevance assessments correlated with other aspects of searching. The most conclusive finding was the number of partially relevant items was often positively correlated with a change in search topic or criteria for relevance: the more partial relevance assessments at a given stage in a search, the more uncertain is the user's current information need.

This study concentrated mainly on users at the initial search stage, when information needs are more likely to be variable. However, partial relevance assessments as an indicator of search stage or search status may be useful in defining what type of documents should be retrieved. For example we may wish to increase retrieval of loosely-related material at certain stages, and suppress retrieval to only highly relevant material at other stages.

A further important factor in determining how users will make relevance assessments is the *task* the user is trying to complete. Users with different tasks will obviously mark different documents relevant, but a user with a long-running task may change their criteria for relevance over time. Spink [Spi96] for example, reports on a study of when and how academics use IR systems over the course of a research project. The majority of users search at the beginning of project and many search again throughout the project. One reason for searching at later stages of projects is to check new updated references - rerunning same searches against new data - but many users modify their search terms over time, either as their information problems change or they obtain information from new sources. Although the searches are similar and the basic topic of the searches are broadly the same, the reasons for searching and the type of information being sought is different leading to different relevance assessments.

[37] In this context a partial assessment means a document is only somewhat relevant to the topic or the user is not sure of the document's relevance. This is distinguished from the situation where only part of the document is relevant.

Vakkari, [Vak00a, Vak00b], also examined long-running searches to examine how relevance assessments changed over time. In his study he demonstrated that not only did subjects chose different documents at different stages in their task, they also used different search tactics and strategies[38]. Vakkari provided support for Spink's observation that high numbers of partial assessments correlates with a lack of ability to discriminate relevant and non-relevant. This may occur at the start of a search, for example. He also found evidence to indicate that when a user has a good idea of what constitutes relevant material he is less likely to make a high number of relevance assessments

These studies are important for RF because they point to the fact that not all relevance assessments are equal: users make assessments for different reasons and with different amounts of knowledge. RF techniques developed so far tend not to make these distinctions or incorporate this kind of knowledge.

8 Conclusion

RF has proved to be a useful and pragmatic solution to the uncertainty of describing an information need. It has further, in test collection evaluations, been shown to be a relatively stable procedure: it works in most cases, a wide range of algorithms give approximately the same performance and how the algorithmic parameters should be set are fairly well understood. Although we have not discussed non-text documents, such as images or speech, in this paper the same basic principle of selecting good discriminators of relevance can be used for different media to implement RF functionality.

The conceptual simplicity of RF – users only have to recognise useful material, not describe it – neatly hides the complexity and variety of the query modification features behind the interface. However, there is a growing awareness that RF is not sufficient on its own to improve retrieval. RF is useful in that it is conceptually simple but it does not yet provide adequate support for the range of strategies and tactics demonstrated by the user in research such as [Bat90]. RF may only be part of the interaction process and will require integration with other functionalities.

Further, although RF is simple for the user to employ, the interaction decisions involved in RF can be obscure. That is, RF generally does not give the user enough context on which to based their relevance decisions, e.g. how many documents should be marked as relevant, how relevant should a document be before being marked as relevant, what does not relevant mean? Although RF research has answers to some of these questions (e.g. more relevance information is generally better), getting the user to provide the necessary input data is not easy, and making the process of assessing relevance more difficult may result in less interaction not more.

Therefore we argue that the strength of RF shown in non-interactive situations should exploited in the interactive situation by paying much more attention to the users of RF techniques and how they incorporate RF into their searching. Finally, we note that RF is not only a potentially useful technique for improving the quality of a searching but is also a very useful for technique for investigating *how* people search. Only by studying how people actually interact with systems can we understand how to build more usable and useful search systems.

Acknowledgements

We would like to express our thanks to Keith van Rijsbergen and Joemon Jose who gave many useful comments on this survey. We would also like to acknowledge the helpful suggestions made by the anonymous reviewer.

References

[Aal92] I. J. Aalbersberg. *Incremental relevance feedback*. Proceedings of the Fifteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp 11-22. Copenhagen. 1992.

[BYRN99] R. Baeza-Yates and B. Ribeiro-Neto. *Modern information retrieval*. Addison-Wesley. 1999.

[38] As measured using Kuhlthau's categorisation of searching, [Kuh91, Kuh93]

[BS98] C. L. Barry and L. Schamber. *Users' criteria for relevance evaluation: a cross-situational comparison*. Information, Processing and Management. **34**. 2/3. pp 219-237. 1998.

[Bat90] M. Bates. *Where should the person stop and the information search interface start?* Information, Processing and Management. **26**. 5. pp 575-592. 1990.

[Bay63] T. Bayes. *An Essay Toward Solving a Problem in the Doctrine of Chances*. Philosophical Transactions of the Royal Society of London. **53**. pp370-418. 1763.

[Beau97] M. Beaulieu. *Experiments with interfaces to support query expansion*. Journal of Documentation. **53**. 1. pp 8-19. 1997.

[BJ98] M. Beaulieu and S. Jones. *Interactive searching and interface issues in the Okapi best match probabilistic retrieval system*. Interacting with computers. **10**. 3. pp 237-248. 1998.

[Bel00] R. K. Belew. *Finding out about*. Cambridge University Press. 2000.

[Bel97] N. J. Belkin. *An overview of results from Rutgers's investigation of interactive information retrieval.* Proceedings of the 34th Annual Clinic on Library Applications of Data Processing. Visualizing Subject Access for 21st Century Information Resources. P. Cochrane (ed). 1997.

[BCK+96] N. J. Belkin, C. Cool, J. Koenemann, K. Bor Ng, S. Park. *Using relevance feedback and ranking in interactive searching*. Proceedings of the Fourth Text Retrieval Conference (TREC-4). (D. Harman ed). NIST Special Publication 500-236. pp 181-210. 1996.

[BCC+97] N. J. Belkin, A. Cabezas, C. Cool, K. Kim, K. B. Ng, S. Park, R. Pressman, S. Rieh, P. Savage, H. Xie. *Rutgers Interactive Track at TREC-5*. Proceedings of the Sixth Text Retrieval Conference (TREC-5). (E.M. Voorhees and D. K. Harman eds). NIST Special Publication 500-238. pp 257-266. 1997.

[BCK+99] N.J. Belkin, J. Perez Carballo, D. Kelly, S. Lin S.Y. Park, S.Y. Rieh, P. Savage-Knepshield, C. Sikora, and C. Cool. *Rutgers' TREC-7 Interactive Track Experience*. Proceedings of the Seventh Text Retrieval Conference (TREC-7). (E.M. Voorhees and D. K. Harman eds). NIST Special Publication 500-242. pp 275-284. 1999.

[BCC98] N. J. Belkin, J. Perez Carballo, C. Cool, S. Lin, S. Y. Park, S. Y. Rieh, P. Savage, C. Sikora, H. Xie and J. Allan. *Rutgers' TREC-6 interactive track experience*. Proceedings of the Sixth Text Retrieval Conference (TREC-6). (E.M. Voorhees and D. K. Harman eds). NIST special publication 500-240. pp 597-610. 1998.

[BKF+95] N. J. Belkin, P. Kantor, E. A. Fox and J. A. Shaw. *Combining the evidence of multiple query representations for information retrieval*. Information Processing and Management. **31**. 3. pp 431-448. 1995.

[BVB97] F. Berger and P. van Bommel. *Augmenting a characterization network with semantic information*. Information Processing and Management. **33**. 4. pp 45 -479. 1997.

[Bha92] S.K. Bhatia. *Selection of search terms based on user profile*. Proceedings of the 1992 ACM/SIGAPP Symposium on Applied computing (vol I): technological challenges of the 1990's. pp 224-233. 1992.

[BI97] P. Borlund and P. Ingwersen. *The development of a method for the evaluation of interactive information retrieval systems.* Journal of Documentation. **53**. 5. pp 225-250. 1997.

[BI99] P. Borlund and P. Ingwersen. *The application of work tasks in connection with the evaluation of interactive information retrieval systems: empirical results*. Mira '99. S. Draper, M. Dunlop, I. Ruthven and C. J. van Rijsbergen (eds). Electronic Workshops in Computing. British Computer Society. 1999.

[Borg89] C. L. Borgman. *All users of information retrieval systems are not created equal: an exploration into individual differences*. Information Processing and Management. **25**. 3. pp 237-251. 1989.

[Borg96] C. L. Borgman. *Why are online catalogs still hard to use?* Journal of the American Society for Information Science. **47**. 7. pp 493-503. 1996.

[BG94] M. Buckland and F. Gey. *The relationship between recall and precision.* Journal of the American Society for Information Science. **45**. 1. pp 12-19. 1994.

[BSA+95] C. Buckley, G. Salton, J. Allan and A. Singhal. *Automatic query expansion using SMART: TREC-3*. Proceedings of the Third Text Retrieval Conference (TREC-3). D. K. Harman (ed). NIST special publication 500-225. pp 69-80. 1995.

[CCH92] J. P. Callan, W.B Croft and S.M. Harding. *The INQUERY retrieval system*. Database and Expert Systems Applications (DEXA). Valencia. pp 78-83. 1992.

[Cam95] I. Campbell. *Supporting information needs by ostensive definition in an adaptive information space*. MIRO '95. electronic Workshops in Computing, Springer Verlag. Ian Ruthven (ed). 1995.

[Cam99] I. Campbell. *Interactive evaluation of the Ostensive Model, using a new test-collection of images with multiple relevance assessments*. Journal of Information Retrieval. **2**, 1, pp 89-114. 1999.

[CVR96] I. Campbell and C. J. van Rijsbergen. *Ostensive model of information needs.* Proceedings of the Second International Conference on Conceptions of Library and Information Science: Integration in Perspective (CoLIS 2). Copenhagen. pp 251-268. 1996.

[CCR71] Y. K. Chang, C. Cirillo and J. Razon. *Evaluation of feedback retrieval using modified freezing, residual collection & test and control groups*. The SMART retrieval system - experiments in automatic document processing. G. Salton (ed). Chapter 17. pp 355-370. 1971.

[CLVR98] F. Crestani, M. Lalmas and C. J. van Rijsbergen. *Information retrieval: uncertainty and logics - Advanced models for the representation and retrieval of information*. Kluwer Academic Publishers. 1998.

[CPB+96] C. Cool, S. Park, N. J. Belkin, J. Koenemann, & K.B. Ng. *Information seeking behavior in new searching environment.* Proceedings of the Second International Conference on Conceptions of Library and Information Science: Integration in Perspective (CoLIS 2). Copenhagen. pp 403-416. 1996.

[CRS+95] F. Crestani, I. Ruthven, M. Sanderson and C. J. van Rijsbergen. *The troubles with using a logical model of IR on a large collection of documents*. Proceedings of the Fourth Text Retrieval Conference (TREC-4). NIST special publication 500-236. D. K. Harman (ed). pp 509-525. 1995.

[CH79] W. Croft and D. Harper. *Using probabilistic models of information retrieval without relevance information*. Journal of Documentation. **35**. 4. pp 285-295. 1979.

[DMB98] S. Dennis, R. McArthur and P. Bruza. *Searching the WWW made easy? The Cognitive Load imposed by Query Refinement Mechanisms*. Proceedings of the Third Australian Document Computing Symposium. 1998.

[DBM97] N. Denos and C. Berrut and M. Mechkour. *An Image Retrieval System based on the Visualization of System Relevance via Documents*. Database and Expert Systems Applications (DEXA). Toulouse. pp 214-224. 1997.

[Dun97] M. D. Dunlop. *The effect of accessing non-matching documents on relevance feedback*. ACM Transactions on Information Systems. **15**. 2. pp 137-153. 1997.

[Dra00] S. Draper. Personal communication.

[Efth93] E. N. Efthimiadis. *A user-centred evaluation of ranking algorithms for interactive query expansion.* Proceedings of the Sixteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp 146-159. Pittsburgh. 1993.

[Efth95] E. N. Efthimiadis. *User-choices: a new yardstick for the evaluation of ranking algorithms for interactive query expansion.* Information processing and management. **31**. 4. pp 605-620. 1995.

[EB88] M. Eisenberg and C. Barry. *Order effects: a study of the possible influence of presentation order on user judgements of document relevance*. Journal of the American Society of Information Science. **39.** 5. pp 293-300. 1988.

[EB94] E. Efthimiadis and P. Biron. *UCLA-Okapi at TREC-2: query expansion experiments*. Proceedings of the Second Text Retrieval Conference (TREC-2). NIST Special Publications 500-215. D. K. Harman (ed). pp 279-290. 1994.

[Ell89] D. Ellis. *A behavioural approach to information system design*. Journal of Documentation. **45**. 3. pp 171-212. 1989.

[FST+99] V. I. Fants, J. Shapiro, I. Taksa and V. G. Voiskunskii. *Boolean search: current state and perspectives*. Journal of the American Society of Information Science. **50**. 1. pp 86-95. 1999.

[Fid91] R. Fidel. *Searchers' selection of search keys: I*. Journal of the American Society of Information Science. **34**. 1991.

[FM95] V. Florance and G. Marchionini. *Information processing in the context of medical care.* Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Seattle. pp 158-163. 1995

[FB00] H. Fowkes and M. Beaulieu. *Interactive searching behaviour: Okapi experiment for TREC-8*. IRSG 2000 Colloquium on IR Research. Cambridge. 2000.

[FMS91] H. P. Frei, S. Meienberg and P. Schauble. *The perils of interpreting recall and precision values.* Information Retrieval. N. Fuhr (ed). pp 1-10. Springer Verlag. 1991.

[FB91] N. Fuhr and C. Buckley. *A probabilistic learning approach for document indexing*. ACM Transactions on Information Systems. **9**. 3. pp 223-248. 1991.

[HC93] D. Haines and W. B. Croft. *Relevance feedback and inference networks.* Proceedings of the Sixteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp 2-11. Pittsburgh. 1993.

[Har88] D. Harman. *Towards interactive query expansion.* Proceedings of the Eleventh Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp 321-331. Grenoble. 1988.

[Har92a] D. Harman. *Ranking algorithms.* Information Retrieval: Data Structures & Algorithms. Englewood Cliffs: Prentice Hall. W.B. Frakes and R. Baeza-Yates (eds). Chapter 14. pp 363-392. 1992.

[Har92b] D Harman. *Relevance feedback revisited.* Proceedings of the Fifteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp 1-10. Copenhagen. 1992.

[Har92c] D. Harman. *Relevance feedback and other query modification techniques.* Information Retrieval: Data Structures & Algorithms. Englewood Cliffs: Prentice Hall. (W.B. Frakes and R. Baeza-Yates ed). Chapter 11. pp 241-263. 1992.

[Har93] D. Harman. *Overview of the first text retrieval conference (TREC-1).* Proceedings of the First Text Retrieval Conference (TREC-1). NIST special publication 500-207. D. K. Harman (ed). pp 1-20. 1993.

[Hea99] M. Hearst. *User interfaces and visualisation*. Modern information retrieval. R. Baeza-Yates and B. Riberio-Nelo (eds). Chapter 10. pp 257-323. Addison-Wesley/ACM Press. 1999.

[Heu99] R. J. Heuer, Jr. *Psychology of Intelligence Analysis*. Center for the Study of Intelligence. Central Intelligence Agency. 1999

[HY93] I. Hsieh-Yee. *Effects of searcher experience and subject knowledge on the search tactics of novice and experienced searchers*. Journal of the American Society for Information Science. **44**. 3. pp 161 - 174. 1993.

[Hui96] T. Huibers. *An axiomatic theory for information retrieval*. PhD thesis. Utrecht University. 1996.

[Ide71] E Ide. *New experiments in relevance feedback.* The SMART retrieval system - experiments in automatic document processing. (G. Salton ed). Chapter 16. pp 337-354. 1971.

[IdS71] E. Ide and G. Salton. *Interactive search strategies and dynamic file organization in information retrieval*. The SMART retrieval system - experiments in automatic document processing. (G. Salton ed). Chapter 18. pp 373-393. 1971.

[Ing92] P. Ingwersen. *Information retrieval interaction*. Taylor-Graham. 1992.

[Ing94] P. Ingwersen. *Polyrepresentation of information needs and semantic entities: elements of a cognitive theory for information retrieval interaction*. Proceedings of the Seventeenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp 101-110. Dublin. 1994.

[Ing96] P. Ingwersen. *Cognitive perspectives of information retrieval interaction: elements of a cognitive IR theory*. Journal of Documentation. **52**. 1. pp 3-50. 1996.

[Iwa00] M. Iwayama. *Relevance feedback with a small number of relevance judgements: incremental relevance feedback vs document clustering*. Proceedings of the twenty-third annual international ACM SIGIR Conference on Research and development in information retrieval. pp 10-16. Athens. 2000.

[JJ91] J. W. Janes. *Relevance judgements and the incremental presentation of document representations*. Information Processing and Management. **27**. 6. pp 629-646. 1991.

[KQC+95] J. Koenemann, R. Quatrain, C. Cool and N. J. Belkin. *New tools and old habits: the interactive searching behavior of expert online searchers using INQUERY*. Proceedings of the Third Text Retrieval Conference (TREC-3). (D. K. Harman ed). NIST Special Publications 500-225. pp 145-178. 1995.

[KP94] C. S. G. Khoo and D. C. C. Poo. *An expert system approach to online catalog subject searching*. Information Processing and Management. **30.** 2. pp 223-238. 1994.

[KB96] J. Koenemann and N. J. Belkin. *A case for interaction: a study of interactive information retrieval behavior and effectiveness*. Proceedings of the Human Factors in Computing Systems Conference (CHI'96). pp 205-212. Zurich. 1996.

[Kuh91] C.C. Kuhlthau. *Inside the search process: information seeking from the user's perspective*. Journal of the American Society of Information Science. **42**. 5. pp 361 - 371. 1991.

[Kuh93] C.C. Kuhlthau. *Principle for uncertainty for information seeking*. Journal of Documentation. **49**. 4. pp 339-355. 1993.

[Lal96] M. Lalmas. *Modelling information retrieval with Dempster-Shafer's theory of evidence: a case study*. ECAI Workshop on Uncertainty in Information Systems: Questions of Viability. pp 29-36. Budapest. 1996.

[LaBr98] M. Lalmas and P.D. Bruza. *The use of logic in information retrieval modelling*. Knowledge Engineering Review. **13**. 2. pp 1-33. 1998.

[Lee98] J. H. Lee. *Combining the evidence of different relevance feedback methods for information retrieval.* Information Processing and Management. **34**. 6. pp 681-691. 1998.

[LGF97] C. Lundquist, D. A. Grossman and O. Frieder. *Improving relevance feedback in the vector space model*. Proceedings of the sixth international conference on Information and knowledge management (CIKM '97). Las Vegas. pp 16-23. 1997.

[MVR97] M. Magennis and C. J. van Rijsbergen. *The potential and actual effectiveness of interactive query expansion*. Proceedings of the Twentieth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp 324-331. Philadelphia. 1997

[Mar64] M. E. Maron. *Mechanized documentation: the logic behind a probabilistic interpretation.* Statistical Association Methods For Mechanized Documentation. National Bureau of Standards Miscellaneous Publications 269. (M. E. Stevens, V. E. Guiliano and L. B. Heilprin. eds). pp 9-13. 1964.

[MK60] M. E. Maron and J. L. Kuhns. *On relevance, probabilistic indexing and information retrieval.* Journal of the Association for Computing Machinery. **15**. pp 8-36. 1960. Reprinted in Readings in Information Retrieval. K. Sparck Jones and P Willet (eds). Morgan Kaufman. pp 39-46. 1997.

[Mar82] W. A. Martin. *Helping the less experienced user*. Proceedings of the 6th International Online Meeting. pp 67-76. 1982.

[MSB98] M. Mitra, A. Singhal and C. Buckley. *Improving automatic query expansion*. Proceedings of the Twenty-First Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp 206-214. Melbourne. 1998.

[Mull98] A. Müller. *A flexible framework for multimedia information retrieval.* Information Retrieval: Uncertainty and Logics - Advanced models for the representation and retrieval of information. (F. Crestani, M. Lalmas and C. J. van Rijsbergen (eds). Kluwer Academic Publishers. Chapter 5. pp 97-127. 1998.

[Nie89] J. Nie. *An information retrieval model based on modal logic.* Information Processing and Management. **25**. 5. pp 471-490. 1989.

[PW91] H. J. Peat and P. Willett. *The limitations of term co-occurrence data for query expansion in document retrieval systems*. Journal of the American Society for Information Science. **42**. 5. pp 378-383. 1991.

[Pet89] T. A. Peters. *When smart people fail: an analysis of the transaction log of an online public access catalog*. The Journal of Academic Librarianship. **15**. 5. pp 267-273. 1989.

[Por80] M. F. Porter. *An algorithm for suffix stripping*. Program. **14**. pp 130-137. 1980.

[PG88] M. Porter and V. Galpin. *Relevance feedback in a public access catalogue for a research library: Muscat at the Scott Polar Research Institute*. Program. **22**. 1. pp 1 - 20. 1988.

[Rob77] S. E. Robertson. *The probability ranking principle in IR*. Journal of Documentation,. **33**. 4. pp 294-304. 1977.

[Rob86] S. E. Robertson. *On relevance weight estimation and query expansion*. Journal of Documentation. **42**. 3. pp 182-188. 1986.

[Rob90] S. E. Robertson. *On term selection for query expansion.* Journal of Documentation. **46**. 4. pp 359-364. 1990.

[RB78] S. E. Robertson and N. J. Belkin. *Ranking in principle*. Journal of Documentation. **34**. 2. pp 93-100. 1978.

[RSJ76] S E Robertson and K Sparck Jones. *Relevance weighting of search terms.* Journal of the American Society for Information Science. **27**. 3. pp 129-146. 1976.

[RW94] S. E. Robertson and S. Walker. *Some simple effective approximations to the 2 Poission model for probabilistic weighted retrieval*. Proceedings of the Seventeenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp 232-241. Dublin. 1994.

[RWH+93] S. E. Robertson, S. Walker, M. Hancock-Beaulieu, A. Gull and M. Lau. *Okapi at TREC*. Proceedings of the First Text Retrieval Conference (TREC-1). NIST special publication 500-207. (D. K. Harman ed). pp 21-30. 1993.

[RWJ+95] S. E. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu and M. Gatford. *Okapi at TREC-3*. Proceedings of the Third Text Retrieval Conference (TREC-3). NIST special publication 500-225. (D. K. Harman ed). pp 109-126. 1995.

[Roc71] J J Rocchio. *Relevance feedback in information retrieval* The SMART retrieval system - experiments in automatic document processing. (G. Salton ed). Chapter 14. pp 313-323. 1971.

[RLVR01] I. Ruthven, M. Lalmas and C. J. van Rijsbergen. *Empirical investigations on query modification using abductive explanations*. Proceedings of the Twenty-Fourth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. New Orleans. 2001.

[Rut02] I. Ruthven. *On the use of explanations as a mediating device for relevance feedback*. Proceedings of the 6th European Conference on Digital Libraries. (ECDL 2002). Rome. 2002.

[RLVR02a] I. Ruthven, M. Lalmas and C.J. van Rijsbergen. *Combining and selecting characteristics of information use*. Journal of the American Society for Information Science and Technology. **53**. 5. pp 378-396. 2002.

[RLVR02b] I. Ruthven, M. Lalmas, and C.J. van Rijsbergen. *Ranking expansion terms using partial and ostensive evidence*. Proceedings of the 4th International Conference on Conceptions of Library and Information Science. CoLIS 4. Seattle. 2002.

[RLVR03] I. Ruthven, M. Lalmas, and C.J. van Rijsbergen. *Incorporating user search behaviour into relevance feedback*. Journal of the American Society for Information Science and Technology. **54**. 6. pp 528-548. 2003.

[RTJ01] I. Ruthven, A. Tombros and J. Jose. *A study on the use of summaries and summary-based query expansion for a question-answering task*.. 23rd BCS European Annual Colloquium on Information Retrieval Research (ECIR '01). Darmstadt. 2001.

[Sal71] G Salton (ed). The SMART retrieval system - experiments in automatic document processing. 1971.

[SB90] G. Salton and C. Buckley. *Improving retrieval performance by relevance feedback.* Journal of the American Society for Information Science. **41**. 4. pp 288-297. 1990.

[Seb94] F. Sebastiani. *A probabilistic terminological logic for modelling information retrieval.* Proceedings of the Seventeenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp 122-130. Dublin. 1994.

[She91] W. Shenouda. *Online bibliographic searching: how end-users modify their search strategies.* Proceedings of the 53rd Annual Meeting of the American Society for Information Science. **26**. pp 3-48. 1991.

[Sim96] B. Simonnot. *Modélisation multi-agents d'un système de recherche d'information multimédia à forte composante vidéo*, (Multi-Agent Modelling of a multimedia information retrieval system for still images and videos collections). PhD thesis. Henri Poincaré University. 1996.

[SVR83] A. Smeaton and C. J. van Rijsbergen. *The retrieval effects of query expansion on a feedback document retrieval system*. The Computer Journal. **26**. 3. pp 239-246. 1983.

[Sme98] A. Smeaton. *Independence of contributing retrieval strategies in data fusion for effective information retrieval.* Proceedings of the 20th BCS-IRSG Colloquium. Springer-Verlag electronic Workshops in Computing. Grenoble. 1998.

[SJ72] K. Sparck Jones. *A statistical interpretation of term specificity and its application in retrieval.* Journal of Documentation. **28**. 1. pp 11-20. 1972

[SJ79] K. Sparck Jones. *Search term relevance weighting given little relevance information*. Journal of Documentation. **35**. 1. pp 30 -48. 1979.

[SSJ+00a] K. Sparck Jones, S. Walker and S. E. Robertson. *A probabilistic model of information retrieval: development and comparative experiments – Part 1*. Information Processing and Management. **36**. 6. pp 779-808. 2000.

[SSJ+00b] K. Sparck Jones, S. Walker and S. E. Robertson. *A probabilistic model of information retrieval: development and comparative experiments – Part 2*. Information Processing and Management. **36**. 6. pp 809-840. 2000.

[SB64] J. Spiegel and E. Bennett. *A modified statistical association procedure for automatic document content analysis and retrieval*. Statistical Association Methods For Mechanized Documentation. National Bureau of Standards Miscellaneous Publications 269. (M. E. Stevens, V. E. Guiliano and L. B. Heilprin. eds). pp 47-60. 1964.

[Spi96] A. Spink. *Multiple search sessions model of end-user behavior: an exploratory study* Journal of the American Society for Information Science. **47**. 8. pp 603-609. 1996.

[Spi97] A. Spink. *Study of interactive feedback during mediated information retrieval*. Journal of the American Society for Information Science. **48**. 5. pp 382-394. 1997.

[SGB98] A. Spink, H. Greisdorf and J. Bateman. *From highly relevant to not relevant: examining different regions of relevance*. Information Processing and Management. **34**. 5. pp 599-621. 1998.

[SS93] A. Spink, and T. Saracevic. *Dynamics of search term selection process in mediated online searching*. Proceedings of the 56th Annual Meeting of the American Society for Information Science, **30**. pp 63-72. 1993.

[SS97] A. Spink and T. Saracevic. *Interaction in information retrieval: selection and effectiveness of search terms*. Journal of the American Society for Information Science. **48**. 8. pp 741-761. 1997.

[SW99] A. Spink, and T. D. Wilson. *Toward a theoretical framework for information retrieval (IR) evaluation in an information seeking context*. Mira '99: Evaluating Information Retrieval. S. Draper, M. Dunlop, I. Ruthven and C. J. van Rijsbergen (eds). electronic Workshops in Computing. 1999.

[Su94] L. T. Su. *The relevance of recall and precision in user evaluation*. Journal of the American Society for Information Science. **45**. 3. pp 207-217. 1994.

[SYA+98] R. G. Sumner, Jr, K. Yang, R. Akers and W. M. Shaw, Jr. *Interactive retrieval using IRIS: TREC-6 experiments*. Proceedings of the Sixth Text Retrieval Conference (TREC-6). (E.M. Voorhees and D. K. Harman eds). NIST special publication 500-240. pp 711-734. 1998.

[TA88] M. A. Tiamiyu and I. Y. Ajiferuke. *A total relevance and document interaction effects model for the evaluation of information retrieval processes*. Information Processing and Management. **24**. 4. pp 391-404. 1988.

[Vak00a] P. Vakkari. *Cognition and changes of search terms and tactics during task performance.* Proceedings of RIAO Conference on Content-Based Multimedia Information Access. Paris. pp 894-907. 2001.

[Vak00b] P. Vakkari. *Relevance and contributing information types of searched documents in task performance*. Proceedings of the twenty-third annual international ACM SIGIR Conference on Research and development in information retrieval. pp 2-9. Athens. 2000.

[VR79] C. J. van Rijsbergen. *Information retrieval*. Butterworths. 2nd edition. 1979.

[VR86] C. J. van Rijsbergen. *A non-classical logic for information retrieval*. The Computer Journal. **29**. 6. pp 48 -485. 1986.

[VRHP81] C J van Rijsbergen, D. Harper and M. Porter. *The selection of good search terms*. Information Processing and Management. **17**. 2. pp 77-91. 1981.

[VH96] E. M. Voorhees and D. Harman. *Overview of the fifth Text REtrieval Conference (TREC-5)*. Proceedings of the 5th Text Retrieval Conference. pp 1-29. NIST Special Publication 500-238. Gaitherburg.1996.

[VH00] E. H. Voorhees and D. Harman. *Overview of the sixth text retrieval conference (TREC-6)*. Information Processing and Management. **36**. 1. pp 3 - 35. 2000.

[WJR02] R. W. White, J. Jose and I. Ruthven. *A task-oriented study on the influencing effects of query-biased summarisation in web searching*. Information Processing and Management. 2002. *to appear.*

[WB95] S. Willie and P. Bruza. *Users' models of the information space: the case for two search models.* Proceedings of the Seventeenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Seattle. pp 205-210. 1995.