Information FILTERING and Information RETRIEVAL: Two Sides of the Same Coin?

Nicholas J. Belkin and W. Bruce Croft

Information filtering is a name used to describe a variety of processes involving the delivery of information to people who need it. Although this term is appearing quite often in popular and technical articles describing applications such as electronic mail, multimedia distributed systems, and electronic office documents, the distinction between filtering and related processes such as retrieval, routing, categorization, and extraction is often not clear. It is only by making that distinction, however, that the specific research issues associated with filtering can be identified and addressed.

A reasonable first step in defining information filtering is to list the typical characteristics or features of this process. The following features are the most commonly mentioned:

- An information filtering system is an information system designed for unstructured or semistructured data. This contrasts with a typical database application that involves very structured data, such as employee records. The notion of structure being used here is not only that the data conforms to a format such as a record type description, but also that the fields of the records consist of simple data types with well-defined meanings. It is possible, for example, to define a database type for a complex document, such as a journal article, but the meaning of the text, figure and table components of that type are much less well-defined than a typical component of an employee record type, such as the salary. Email messages are an example of semistructured data in that they have well-defined header fields and an unstructured text body.

- Information filtering systems deal primarily with textual information. In fact, unstructured data is often used as a synonym for textual data. It is, however, more general than that and should include other types of data such as images, voice, and video that are part of multimedia information systems. None of these data types are handled well by conventional database systems, and all have meanings that are difficult to represent.
- Filtering systems involve large amounts of data. Typical applications would deal with gigabytes of text, or much larger amounts of other media.
- Filtering applications typically involve streams of incoming data, either being broadcast by remote sources (such as newswire services), or sent directly by other sources (email). Filtering has also been used to describe the process of accessing and retrieving information from remote databases, in which case the incoming data is the result of the database searches. This scenario is also used by the developers of systems that generate "intelligent agents" for searching remote, heterogeneous databases.
- Filtering is based on descriptions of individual or group information preferences, often called profiles. Such profiles typically represent long-term interests.
- Filtering is often meant to imply the removal of data from an incoming stream, rather than finding data in that stream. In the first case, the users of the system see what is left after the data is removed; in the latter case, they see the data that is extracted. A common example of the first approach is an email filter designed to remove "junk" mail. Note that this means profiles may not only express what people want, but also what they do not want.

This list of features suggests that information filtering is a well-defined and unique process. On closer examination, however, many of these features are virtually the same as those found in a variety of other text-based information systems. Text routing, for example, involves sending relevant incoming data to individuals or groups. This process is essentially identical to filtering. Categorization systems [11] are designed to attach one or more predefined categories to incoming objects (this is done by newswire services, for example). The major difference from filtering in this case is the static nature of the categories, when compared to profiles. Extraction systems [27] are somewhat different in that they emphasize the extraction of facts from the text of incoming objects, with the determination of which objects are relevant being a secondary issue. Information retrieval systems [22] share many of the features of information filtering. Indeed, Selective Dissemination of Information (SDI) [14], one of the original functions of information retrieval systems, appears to be identical to most information filtering applications.

A deeper understanding of the differences between filtering and other text-based processes, together with a definition of the research issues involved, requires a more detailed comparison. This comparison, which is the subject of this article, will be based on models of information retrieval developed over the past 20 years of research in this field. We will develop a similar model for information filtering, and compare these models to define research issues. By clarifying the similarities and differences between filtering and retrieval, developers of filtering systems should be able to benefit from the results obtained in related retrieval experiments.

Models of Information Retrieval and Filtering

General Concepts of Information Retrieval and Information Filtering

Information retrieval (IR) has been characterized in a variety of ways, ranging from a description of its goals, to relatively abstract models of its components and processes. Although not all of these characterizations have been in agreement with one another, they all tend to share some commonalities. Usually, an IR system is considered to have the function of "leading the user to those documents that will best enable him/her to satisfy his/her need for information" [17]. Somewhat more generally, "the goal of an information [retrieval] system is for the user to obtain information from the knowledge resource which helps her/him in problem management" [1]. Such functions, or goals, of IR have been described in models of the type shown in Figure 1. This model indicates basic entities and processes in the IR situation.

In this model, a person with some goals and intentions related to, for instance, a work task, finds that these goals cannot be attained because the person's resources or knowledge are somehow inadequate. A characteristic of such a "problematic situation" [23] is an *anomalous state of knowledge* (ASK) [2] or *information need,* which prompts the person to engage in active information-seeking behavior, such as submitting a *query* to an IR system. The query, which must be expressed in a language understood by the system, is a representation of the information need. This is shown on the right-hand side of Figure 1. Due to the inherent difficulty of representing ASKs [2], the query in an IR system is always regarded as approximate and imperfect.

On the other side of Figure 1, the focus of attention is the information resources that the user of the IR system will eventually access. Here, the model considers the *producers* or authors of texts*; the groupings of texts into *collections* (e.g., databases); the *representation* of texts; and, the *organization* of these representations into databases of *text surrogates.* The process of representing the meaning of texts in a form more amenable to processing by computer (sometimes called *indexing*) is of central importance in IR. A typical surrogate would consist of a set of *index terms* or keywords.

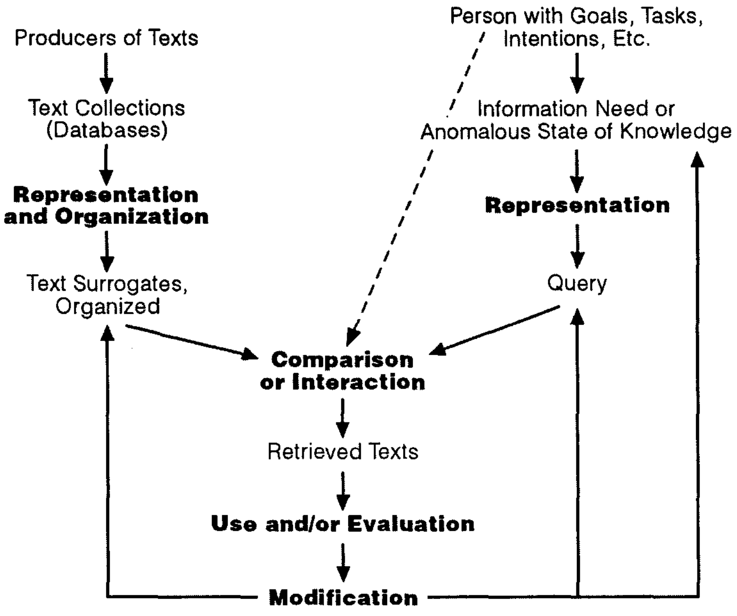
The *comparison* of a query and surrogates, or, in some cases, direct interaction between the user and the

*We use *text* here as a general term that could also include multimedia objects.

texts or surrogates (as in hypertext systems), leads to the selection of possibly relevant *retrieved texts*. These retrieved texts are then *evaluated* or used, and either the user will leave the IR system, or the evaluation leads to some *modification* of the query, the information need, or, more rarely, the surrogates. The process of query modification through user evaluation is known as *relevance feedback* in IR [22].

Research in IR has not considered all of the entities and processes shown in Figure 1 with equal interest. There have been, for instance, almost no studies about the generation of texts, or of their producers, and studies of the collection process have been done almost solely in operational terms. There has been much experimental research in IR that has concentrated on the processes of text representation and organization, comparison, and query modification. This research has been concerned primarily with evaluation of system performance, as measured by precision and recall. Another line of IR research has emphasized studies of the people involved in IR systems, and has investigated issues such as how users get from goals or information needs to queries; representation of states of knowledge underlying queries; the interactive processes in IR, in particular, between users and human intermediaries; the evaluation of texts with respect to a user's tasks and goals; and alternative performance measures for interactive systems.

Based on the general model of IR in Figure 1, and the previous description of information filtering features, a model of information filtering that appears to describe the major entities and processes involved is presented in Figure 2. In this model, information filtering begins with people (the *users* of the filtering system) who have relatively stable, long-term, or periodic *goals* or *desires* (e.g., accomplishing a work task, or being entertained). Groups, as well as individuals, can be characterized by such goals. These then lead to *regular information interests* (e.g., keeping up-to-date on a topic) that may change slowly over time as conditions, goals, and knowledge change. Such infor-


Producers of Texts
Text Collections (Databases)
Representation and Organization
Text Surrogates, Organized
Person with Goals, Tasks, Intentions, Etc.
Information Need or Anomalous State of Knowledge
Representation
Query
Comparison or Interaction
Retrieved Texts
Use and/or Evaluation
Modification

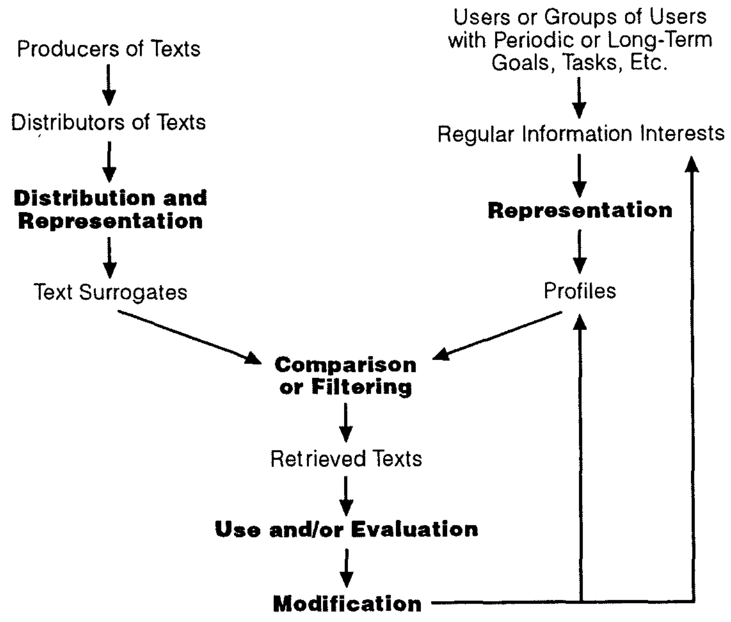


Figure 1. A general model of information retrieval

Figure 2. A general model of information filtering

mation interests lead the people to engage in relatively passive forms of information-seeking behavior, such as having texts brought to their attention. This is accomplished by *representation* of the information interests as *profiles* or *queries* that can be put to the filtering system. Such profiles have generally been construed as good specifications of the information interests.

On the left side of Figure 2, the focus is on *producers* of *texts,* who are often institutions, such as newspapers, as well as individuals. These institutions, or others, such as newsgroups, undertake to *distribute* the texts as they are generated, so they can be brought to users' attention. To accomplish this, the texts are *represented* and *compared* to the profiles. The comparison results in some of the texts being brought to the users' attention (being retrieved). These texts are *used* (or not) and are *evaluated* in terms of how well they respond to the information interests and their motivating goals. The evaluation may lead to *modification* of the profiles and information interests. The modified entities are used in subsequent comparison processes.

In comparing and discussing Figures 1 and 2, we note that at this rather abstract level the entities and processes relevant to information filtering are almost identical to those that are relevant to IR. The major differences appear to be:

- Where IR is typically concerned with single uses of the system, by a person with a one-time goal and one-time query, information filtering is concerned with repeated uses of the system, by a person or persons with long-term goals or interests.
- Where IR recognizes inherent problems in the adequacy of queries as representations of information needs, filtering assumes that profiles can be correct specifications of information interests.
- Where IR is concerned with the collection and organization of texts, filtering is concerned with the distribution of texts to groups or individuals.
- Where IR is typically concerned with the selection of texts from a relatively static database, filtering is mainly concerned with selection or elimination of texts from a dynamic datastream.
- Where IR is concerned with responding to the user's interaction with texts within a single information-seeking episode, filtering is concerned with long-term changes over a series of information-seeking episodes.

In addition to these distinctions based on the models of IR and filtering, there seem to be some other, *contextual* differences that might also be relevant to research interests. These arise from differences in the social and/or practical situations with which IR and filtering have been concerned. Such differences could be categorized according to differences associated with the texts, the users, and the general environment of concern to each.

- Text-related issues. For information filtering, the *timeliness* of a text is often of overriding significance. For IR, this has typically not been the case.
- User-related issues. IR has, by-and-large, studied well-defined user groups, in well-defined, specific domains, largely in science and technology. These users have almost always been highly motivated in their information-seeking behaviors. Filtering, however, is often concerned with very undefined user communities, such as people seeking entertainment in their homes, and with highly varied domains. Also, motivation in the filtering environment cannot always be assumed.
- Environmental issues. Here, the most salient difference seems to be that filtering is highly concerned, in many situations, with issues of privacy; IR, for a variety of reasons, has paid almost no attention to this kind of problem.

Specific Models of Information Retrieval

Having discussed the strong similarities between IR and information filtering in terms of processes such as representation, comparison, and modification, we shall conclude this section with a brief overview of the more specific models that have been developed in IR. These models are primarily focused on the comparison process. The three major alternatives are the Boolean, vector space and probabilistic retrieval models. The first of these is based on what is called the "exact match" principle; the other two on the concept of "best match." For a detailed review, see [2, 22].

Boolean retrieval is based on the concept of an exact match of a query specification with one or more text surrogates. The term "Boolean" is used because the query specifications are expressed as words or phrases, combined using the standard operators of Boolean logic. In this retrieval model, all surrogates, or more generally, texts, containing the combination of words or phrases specified in the query are retrieved, and there is no distinction made between any of the retrieved documents. Thus, the result of the comparison operation in Boolean retrieval is a partition of the database into a set of retrieved documents, and a set of not-retrieved documents.

The Boolean, exact-match retrieval model is the standard model for current large-scale, operational information retrieval systems. A major problem with this model is that it does not allow for any form of relevance ranking of the retrieved document set. That is, it is clear that some texts are more likely to be relevant (or are more relevant) to an information need than others. Presenting documents to the user in presumed order of relevance results in more effective and usable systems. Similarly, excluding documents that do not precisely match a query specification results in lower effectiveness [21, 30].

Best-match retrieval models have been proposed in response to the problems of exact-match retrieval. The most widely known of these is the vector space model [22]. This model treats texts and queries as vectors in a multidimensional space, the dimensions of which are the words used to represent the texts. Queries and texts are compared by comparing the vectors, using, for example, the cosine correlation similarity measure. The assumption is that the more similar a vector representing a text is to a query vector, the more likely that the text is relevant to that query. In this model, an important refinement is that the terms (or dimensions) of a query, or text representation, can be *weighted,* to take account of their importance. These weights are computed on the basis of the statistical distributions of the terms in the database, and in the texts.

Probabilistic information retrieval models are based on the Probability Ranking Principle [16]. This states that the function of an information retrieval system is to rank the texts in the database in the order of their probability of relevance to the query, given all the evidence available. This principle takes into account that representation of both information need and text is uncertain, and the relevance relationship between them is also uncertain. The probabilistic retrieval model suggests there is a variety of sources of evidence that could be used to estimate the probability of relevance of a text to a query. The most typical source of such evidence is the statistical distribution of terms in the database, and in relevant and nonrelevant texts. The next section contains a detailed discussion of a probabilistic retrieval model and how it could be applied to filtering.

It should be noted that both of the best-match models mentioned here can rank documents using Boolean queries [21, 30]. The distinction between the form of the query and the underlying retrieval model is an important one.

Probabilistic Models of Retrieval and Filtering

Filtering in the context of a specific probabilistic retrieval model and an implementation of that model will be discussed in this section. The *inference net* model used for this purpose has been shown to be general, in that it can be used to describe other well-known approaches to retrieval, and effective, in that implementations of the model achieve high levels of recall and precision relative to other systems [30, 31]. The inference net model also allows for a great deal of flexibility in formulating a query and relating the query concepts to the concepts used to describe objects [6].

The Retrieval Model

Probabilistic retrieval models compute P(I|Object), which is the probability that a user's information need is satisfied given a particular object. Objects are usually considered to contain text, although in the context of complex object retrieval, this is often not the case. Our concern in this article shall be mainly with text, although we shall retain the term "object" to indicate that the models are more general. We consider an information need as a complex proposition about the content of an object, with possible values true and false. Queries are regarded as representations of the information need. The major difference between the inference net model and other probabilistic models is that the inference net model emphasizes the use of multiple sources of evidence to calculate P(I|Object).

The inference net model is based on Bayesian inference networks [15]. These are directed, acyclic dependency graphs in which nodes represent propositional variables or constants and edges represent dependence relations between propositions. If a proposition represented by a node p "causes" or implies the proposition represented by node q, we draw a directed edge from p to q. The node q contains a matrix (a *link* matrix) that specifies $P(q|p)$ for all possible values of the two variables. When a node has multiple parents, the matrix specifies the dependence of that node on the set of parents and characterizes the dependence relationship between that node and all nodes representing its potential causes. Given a set of prior probabilities for the roots of the network, these networks can be used to compute the probability or degree of belief associated with all remaining nodes.

Figure 3 shows the basic inference network used in this article. The network consists of an object network and a query network. The object network is built once for a collection and its structure does not change during query processing. The query network consists of a single node representing the user's information need and one or more query representations expressing that information need. A query network is built for each information need and is modified through interactive query formulation or relevance feedback.

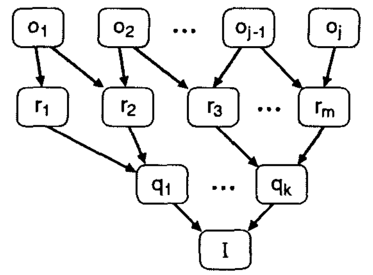
The object network consists of object nodes (o_j's) and concept representation nodes (r_m's). We represent the assignment of a specific representation concept to an object by a directed arc to the representation node from each node representing an object to which the concept has been assigned. A representation node contains a specification of the conditional probability associated with the node, given its set of parent object nodes. Representation nodes are generated through indexing, either automatic or manual. In a typical information retrieval system, they will correspond to words extracted from the text [22], although representations based on more sophisticated language analysis are also possible. The estimation of the probabilities $P(r_m|o_j)$ is based on the occurrence frequencies of concepts in both individual objects and large collections of objects.

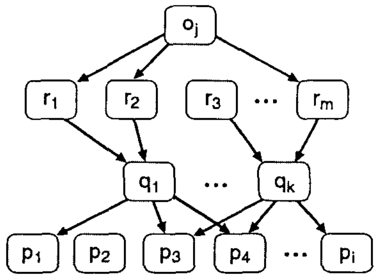
The query network contains a single node (I) corresponding to the event that an information need is met and multiple roots (q_k's) corresponding to the concepts that express the information need. A set of intermediate query nodes may be used to describe complex query networks, such as those formed with Boolean expressions [6].

For retrieval, a query network is built through interaction with the user, and attached to the object network. This allows us to compute the probability that the information need is met for any particular object and, consequently, to produce a ranked list of objects. More details of this process can be found in [30].

The Filtering Model

Given the description of the retrieval model in the previous subsection, we can now describe a similar model for information filtering that attempts to incorporate the characteristic features mentioned earlier in the article. Figure 4 shows the structure of this model. The differences between this model and the retrieval model in Figure 3 reflect the fact that, in filtering, an incoming stream of objects is compared to many profiles at the same time, rather than a single query





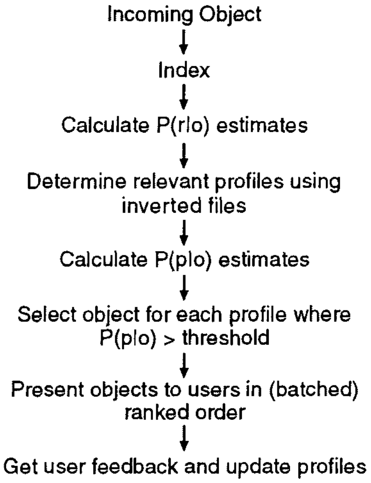

Incoming Object
Index
Calculate P(r|o) estimates
Determine relevant profiles using inverted files
Calculate P(p|o) estimates
Select object for each profile where P(p|o) > threshold
Present objects to users in (batched) ranked order
Get user feedback and update profiles

Figure 3. Basic inference network: o_j's are object nodes, r_m's are concept nodes, q_k's are query nodes, and I represents the user's information need.

Figure 4. Inference network for filtering: o_j is the node associated with the incoming object, r_m's are concept nodes, q_k's are query nodes, and p_i nodes represent the profiles.

Figure 5. A filtering process based on the inference net

being compared to a large, relatively static database. Conceptually, this means that, for every incoming object o_j, we compute the probabilities associated with all profile nodes p_1 through p_n. Based on that computation, we "filter" the object, which may mean removing the object from the stream for a given profile or selecting an object for a profile, depending on the application. This filtering model raises many more detailed issues, however, that must be addressed in order to build filtering systems.

These issues can be clarified by considering a definition of filtering in the context of the probabilistic model. Given a particular object from the incoming stream of objects and a set of profiles, what exactly does it mean to "filter" that object? From an intuitive point of view, it would seem reasonable to select the best-matching profiles for the object. This, however, is too simple to serve as a general model. The inference net model describes how to calculate the probability that a given profile (representing an information need) is true given in the incoming object. In the case of retrieval, this probability is used to rank objects for presentation to the user. This situation would only occur in filtering, however, if we make the simplifying assumption that incoming objects are batched together and ranked relative to each profile. Filtering in this case becomes a minor variation of retrieval, and it results in all incoming objects being presented (in different rank orders) to the users associated with every profile. Although this may be feasible for some applications, there are many in which this batching of incoming objects would not be possible.

If we do not rank incoming objects in batches, but instead must decide on the relevance of each object as it appears, then there are a number of possibilities. We could, for example, direct an object to the users associated with the top-ranking set of profiles. The problem with this approach is that we must choose some fixed number of profiles from the top of the ranking, without regard to how well the profiles matched the object. Alternatively, we could attempt to set a threshold on how similar an object must be to a profile. A more formal definition of this threshold comes from interpreting the inference net model as a Bayesian decision model. This means we decide that an object o_j is relevant to a profile p_i if $P(p_i \textit{ is true}|o_j) > P(p_i \textit{ is false}|o_j)$, assuming that the costs of decision errors are equal [10]. The problem of setting the threshold then becomes the more general problem of obtaining accurate probability estimates.

In general, then, filtering could be defined as the process of determining which profiles have a high probability of being satisfied by a particular object from the incoming stream. Objects with low probabilities for a particular profile are removed from the stream of objects directed to the users associated with that profile. Objects in that stream could be batched and presented in ranked order using the probabilities, if that is appropriate for the application.

This model can handle "negative" profiles straightforwardly. These profiles describe the features of objects that are not wanted, rather than the features that are wanted. Objects that do not contain these features have high probabilities of satisfying the profile and will not be removed.

The implementation of a filtering system based on this model involves two main conceptual issues and a number of efficiency problems. The first issue is related to indexing, or representing the contents of objects. The indexing process in a text-based filtering system will be essentially the same as in a text retrieval system, especially a system that deals with heterogeneous databases. In order to handle the many different formats of the objects and the dynamic nature of the language in those objects, it is necessary to use fairly simple word- and phrase-based indexing techniques [22]. It is important to realize, however, that the representation of the information need is not limited to these simple features. More complex features can be constructed from these features using, for example, Boolean operators [30], phrase-recognition techniques [6], and rules [28]. These complex features can be modeled directly in the inference net framework. It would also be possible to recognize these features using a more sophisticated indexing process. In the context of filtering, however, where a large incoming stream of documents may need to be indexed very quickly, the retrieval effectiveness benefits obtained from improved indexing must be balanced against the loss of indexing efficiency.

The issue of probability estimation is a major one in any retrieval system (in some systems, the probabilities are "weights"). In a filtering system, the problem is worse in some respects and better in others. The problem is worse because objects arrive in streams rather than being available as static databases. The estimation of the indexing probabilities ($P(r_m|o_j)$ in Figure 3) is done using word and phrase frequencies in the individual object text and in the database of objects. To obtain accurate estimates for the probabilities based on the "universe" of objects, it is necessary to base those estimates on large samples of objects seen previously. It may even be necessary to maintain these sample probabilities for each of the sources of objects for the filtering system.

Estimating the probabilities in the query (or profile) network in a filtering system is easier than in a retrieval system because of the long-term nature of the associated information needs. In this situation, there are likely to be many more examples of objects that satisfied the profiles, and therefore there is more opportunity to learn the correct probabilities. Relevance feedback techniques used in retrieval systems [22] generally improve the retrieval effectiveness significantly and they are even more likely to do so in a filtering system [13].

In terms of efficiency, the main problem is that retrieval systems are typically implemented using inverted files of document representatives. In the case of the inference net model, the probabilities $P(r_m|o_j)$ in the object network are precomputed and stored in inverted lists, one for each concept [29]. This is a very efficient approach when there are many objects to be compared to a single query. For a filtering system, however, we will often be comparing a single object to a large number (perhaps thousands) of profiles, so it is unlikely that the same implementation will suffice. Instead, each incoming object could be indexed and have the associated probabilities calculated at filtering time. These probabilities could then used to evaluate profile networks containing the features present in the object. To determine which profiles satisfy that constraint, assuming there are large numbers of profiles, inverted lists of query concepts could be constructed.

The filtering process suggested by the model introduced in this section is summarized in Figure 5. We believe this process could be used to describe most of the filtering applications that have been suggested. In addition, the filtering model clarifies the assumptions and issues that underlie such applications.

A final point to note is that, unlike simpler models such as the vector-space model [22], objects and profiles are not symmetric in the inference net model. By this, we mean that we cannot simply turn the inference net "upside down" to make the model in Figure 4 look more like that in Figure 3. We cannot do this because we do not really understand what the probability $P(o_j|p_i)$ means or how to compute it. The information need is never "observed," since it is inside peoples' heads. Although this makes our filtering models somewhat more complicated, we believe that the probabilistic approach results in a better understanding of the key issues and new approaches to addressing them.

Lessons for Filtering from Retrieval Research

Given that a number of components of a text-based filtering system will be virtually identical to those in a text retrieval system, it is reasonable to ask what has been learned from experiments with text retrieval systems, and how do those results apply to a filtering system. Research in IR can be classified into the three main categories mentioned earlier in this article, and we will base our discussion of this research on them. The categories of research are text representation, retrieval (comparison) techniques, and acquisition of information needs.

Text Representation

Text representation, or indexing, has been one of the major foci of research in IR [12, 18, 22, 25]. The result that is most important to filtering is that simple word-based representations, when combined with appropriate retrieval models, are surprisingly effective as well as being efficient and straightforward to implement. Indexing an object for filtering using this approach consists of lexical scanning to identify words, morphological analysis to reduce different word forms to common "stems," and counting occurrences of those stems. The simplicity of this process means that probabilistic approaches to filtering are feasible even with very high volumes of incoming objects. An extension of this indexing process that is very useful for some applications is to include special-purpose recognizers in the scanner. Some important types of features that could be recognized in this way are company names, peoples' names, dates, and locations.

More sophisticated representations based on natural language processing techniques have yet to be shown to be cost-beneficial. This includes even simple techniques such as recognizing noun phrases using syntactic or stochastic parsing. Although there is some evidence that the recognition of phrases in queries using these techniques is effective [6], the importance of a phrase-based concept in an object can be generally identified using simple word proximity measures. Despite the difficulty of making progress in this area, the recent upsurge in interest in large-scale applications of natural language processing holds promise for eventually improving the effectiveness of filtering systems. The research on text extraction carried out under the DARPA-sponsored Message Understanding and Evaluation Conference [27], in particular, indicates that advanced techniques can be used to extract specific information from text and could provide more accurate evidence for the relevance of text objects. The DARPA TIPSTER program is continuing this research, and is also undertaking the first large-scale evaluations of filtering techniques.

Another representation technique that has been extensively studied in IR is clustering [33]. Document clustering is used to group documents with related representations and term clustering is used to group related words and phrases. In the case of document clusters, representatives of the clusters are used for comparison to the query, rather than the original text representations [24]. The technique can be regarded, therefore, as transforming the original representations. Term clusters, on the other hand, are typically used to expand (or transform) the original query representation. The experiments that have been carried out using these techniques have not established their effectiveness, although a recent application of factor analysis [7] has some promise.

Retrieval Techniques

The use of retrieval models as a basis for retrieval techniques has been discussed earlier in this article. The most important results in the IR literature in this area have to do with the relative effectiveness of different retrieval techniques and probability estimation functions.

Given that ranking techniques should be used to achieve good effectiveness, a basic issue is how the "score" of an object should be calculated. In probabilistic retrieval models, this involves estimating probabilities. In the vector-space model, term weights can be interpreted as probability estimates [31] and a great deal of experimental work has been done to evaluate alternative forms [19]. In general, these are referred to as *tf.idf* weights, since they include a component based on the frequency of a word (or feature) in the text of an object (the term frequency component or *tf*), and a component based on frequency of the word in the "universe" of objects (the inverse document frequency or *idf*). The *idf* weight increases as the frequency of the word decreases (hence the name). The retrieval system based on the inference net model also uses a form of *tf.idf* weight for estimation of the $P(r_m|o_j)$ values [30]. For a filtering system to be effective, it is important that similar estimation functions are used.

Acquisition of Information Needs

Acquiring accurate descriptions of information needs is essential in a retrieval system, and will be just as crucial in a filtering system. As mentioned previously, the profiles in a filtering system often represent long-term interests, and there may be more opportunities to improve the quality of the profile. The research in IR that is relevant to this aspect of filtering has been in query formulation and relevance feedback.

Research in query formulation has focused on query languages and interactive aids to formulation. It has been shown, for example, that Boolean queries are extremely difficult to generate [4]. It has also been shown that Boolean or structured queries can be very effective when used with an appropriate retrieval model [6, 21]. The additional structure in Boolean queries (compared to queries expressed as sets of terms) can describe important linguistic features such as phrases. This suggests that the filtering model should be able to handle structured queries and that interfaces should be designed to support structured query formulation.

It has been shown that user input about concepts related to those mentioned in an initial query, together with their relative importance, can significantly improve retrieval effectiveness [5]. Conversely, other experiments have shown that expanding queries by having users select additional concepts from lists suggested by the system is often not effective [8]. The reasons for these differences are not clear, although it appears that using only system suggestions is too restrictive and does not make full use of the user's domain knowledge. The design of interfaces for filtering systems, therefore, is not straightforward, and the primary components should involve encouraging users to be as specific as possible without limiting them to a choice from a list of topics. One possible approach is to ask users for natural language descriptions of interests, analyze these descriptions using simple natural language processing techniques to isolate concepts, prompt users to supply concepts related to those in the initial statement and to indicate which concepts are related. Systems in which users are expected to supply much more sophisticated descriptions of information needs [28] are limited to the small number of applications where this expectation is reasonable.

The research on relevance feedback has shown that significant effectiveness improvements can be gained by using quite simple feedback techniques [20]. There have also been results showing that the problem of choosing new terms from relevant documents to add to queries becomes worse in full text collections and in applications where large numbers of relevant documents are available to train the system [13]. Techniques that have been effective for feature selection in situations having small numbers of abstract length documents do not appear to be sufficiently discriminating when used to select from thousands of possible features. This means that although feedback is a necessary component of a filtering system, more research is necessary to identify the most appropriate feedback techniques for this task. Relevance feedback can be improved if users select features from the texts of relevant documents [5], but not from lists of terms selected automatically from relevant documents.

Relevance feedback focuses on training the system to respond to a particular profile. It also appears possible to learn probability estimation functions (especially that used to estimate $P(r_m|o_j)$) from the results of many profile-object comparisons [9]. This is particularly interesting for filtering, given the large amount of training data (relevance judgments) that will typically be available.

Evaluation

The field of IR has devoted considerable attention to the issue of evaluation [22, 32]. The distinction between the efficiency and effectiveness of a retrieval system was made early, and the emphasis has been on measuring effectiveness. A number of measures have been developed, with the best-known being recall and precision. Precision is the proportion of a retrieved set of documents that is actually relevant. Recall is the proportion of all relevant documents that are actually retrieved. These figures are typically presented as averages over sets of queries.

In many filtering applications, recall and precision will be adequate for evaluating effectiveness. It has been pointed out, however, that evaluating a filtering system's performance at selecting the right profiles in response to incoming documents can require variations of the standard measures [11]. One example of the difference is that in a filtering system, each incoming document may have to be assigned to a subset of the current profiles, whereas in the retrieval context, the assignment does not have to be made because all documents are ranked for each query. The concern with establishing ranking thresholds to determine assignments to profiles results, at the very least, in different averaging techniques being used in the evaluation.

There is also concern being expressed in the IR community over the value and validity of the standard recall and precision measures in interactive contexts [26]. Researchers doing experiments with information filtering will be able to benefit from the long IR experience with evaluation, but the development of criteria, measures and methods tailored to the evaluation of filtering systems is an important issue that will also have an impact on IR research.

Conclusion

We began this article by considering the relationship between information filtering and information retrieval. It seems fair to say, after having examined the foundations of each of these enterprises, that there is relatively little difference between the two, at an abstract level. First of all, their underlying goals are essentially equivalent. That is, both are concerned with getting information to people who need it, and both are concerned with more-or-less the same kind of information, and the same kind of context. Furthermore, most of the issues which appear at first to be unique to information filtering, are really specializations of IR problems. The extended discussion of the probabilistic inference net approach to IR, and its application to information filtering, seems to demonstrate this relationship rather concretely. The conclusion we draw from this is that much of IR research experience is directly relevant to filtering.

It is clear, however, that IR research has ignored some aspects of the general problem to which both IR and information filtering address themselves, and these are precisely the aspects which are especially relevant to the specific contexts of filtering. The following is a summary of specific issues that have been discussed in previous sections of this article.

Learning and adaptation are issues that have been of concern to IR research, primarily through the concept of relevance feedback. However, such research has been based on relatively meager training sets, and applied in fairly small databases. Information filtering is concerned with much larger data sets, and, generally, with information needs which are relatively stable over relatively long periods of time.

There has been relatively little experience with the indexing of nontextual data in IR. Information filtering, in many of its contexts, is crucially concerned with multimedia texts. Although interest in this problem is converging for both fields, it seems likely that this will be a more important research issue for information filtering than for IR in the near-term future.

The timeliness of data is another area of particular concern to filtering. Research is needed on how to represent temporal constraints, how to understand when a text is likely to be timely for a particular user, and what timeliness means in specific contexts.

Researchers studying filtering also need to do a great deal of research on the dimensions of users' information interests: what they might be, how to identify them, how to represent them, and how to modify them. This is especially the case because filtering is considering new classes of users, uses and data, for which IR does not, in general, have relevant results. The study of the uses that people make of texts, and the characteristics of texts that are salient to those uses, will be of major concern in the context of information filtering. In particular, applications such as the recreational use of television programming pose special problems and opportunities for research in filtering.

Finally, information filtering clearly involves many economic and social issues, associated with the production and distribution of texts, that have been of relatively little interest to IR. Research in this area is likely to focus on issues pertaining to privacy, copyright, and access.

Thus, it seems there is indeed a research agenda for filtering beyond that which has been charted by IR. While this agenda has much to do with the contexts in which filtering is likely to take place, and its applications, it is also based on the underlying model of what it wants to do. That model, although in many respects equivalent to models of IR, specifically extends it in some interesting and important ways. This extension, and the research agenda accompanying it, seems likely to be of significance to IR as well as filtering, since it addresses issues that should be of importance to IR, but which IR has not addressed, primarily because of specialization to specific contexts and users.

We conclude that information retrieval and information filtering are indeed two sides of the same coin. They work together to help people get the information needed to perform their tasks.

References

1. Belkin, N.J. Cognitive models and information transfer. *Soc. Sci. Inf. Stud. 4* (1984), 111–129.
2. Belkin, N.J. and Croft, W.B. Retrieval techniques. In *Annual Review of Information Science and Technology,* M.E. Williams, Ed. Chapt. 4, pp. 109–145. Elsevier, 1987.
3. Belkin, N.J., Oddy, R.N. and Brooks, H.M. ASK for information retrieval: Part I. Background and theory. *J. Doc. 38,* 2 (June 1982), 61–71.
4. Borgman, C.L. All users of information retrieval systems are not created equal: An exploration into individual differences. *Inf. Process. Manage. 25,* 3 (1989), 237–251.
5. Croft, W.B. and Das, R. Experiments with query acquisition and use in document retrieval systems. In *Proceedings of the ACM SIGIR Conference on*

Research and Development in Information Retrieval, (1990), pp. 349–368.

6. Croft, W.B., Turtle, H.R. and Lewis, D.D. The use of phrases and structured queries in information retrieval. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval,* (1991), pp. 32–45.
7. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K. and Harshman, R. Indexing by latent semantic analysis. *J. Am. Soc. Inf. Sci. 41* (1990), 391–407.
8. Ekmekcioglu, F.C., Robertson, A.M. and Willett, P. Effectiveness of query expansion in ranked-output document retrieval systems. *J. Inf. Sci. 18* (1992), 139–147.
9. Fuhr, N. and Buckley, C. Probabilistic document indexing from relevance feedback data. In *Proceedings of the Thirteenth International Conference on Research and Development in Information Retrieval,* Jean-Luc Vidick, Ed. ACM, New York, Sept. 1990, pp. 45–61.
10. Fukunaga, K. Ed. *Introduction to Statistical Pattern Recognition.* Academic Press, 1990.
11. Lewis, D.D. An evaluation of phrasal and clustered representations on a text categorization task. In *Proceedings of the Fifteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (1992), pp. 37–50.
12. Lewis, D., Croft, W.B. and Bhandaru, N. Language-oriented information retrieval. *Int. J. Intell. Syst. 4* (1989), 285–318.
13. Lewis, D.D. Representation and Learning in Information Retrieval. Ph.D. dissertation, University of Massachusetts at Amherst, 1992.
14. Packer, K.H. and Soergel, D. The importance of sdi for current awareness in fields with severe scatter of information. *J. Am. Soc. Inf. Sci. 30,* 3 (1979), 125–135.
15. Pearl, J. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann, 1988.
16. Robertson, S.E. The probability ranking principle in IR. *J. Doc. 33,* 4 (Dec. 1977), 294–304.
17. Robertson, S.E. The methodology of information retrieval experiment. *Information Retrieval Experiment.* In K. Sparck Jones, Ed. Chapt. 1, pp. 9–31. Butterworths, 1981.
18. Salton, G. Another look at automatic text-retrieval systems. *Commun. ACM 29,* 7 (July 1986), 648–656.
19. Salton, G. and Buckley, C. Term weighting approaches in automatic text retrieval. *Inf. Process. Manage. 24,* 3 (1988), 513–524.
20. Salton, G. and Buckley, C. Improving retrieval performance by relevance feedback. *JASIS 41* (1990), 288–297.
21. Salton, G., Fox, E. and Wu, H. Extended Boolean information retrieval. *Commun. ACM 26,* 11 (Nov. 1983), 1022–1036.
22. Salton, G. and McGill, M.J. *Introduction to Modern Information Retrieval.* McGraw-Hill, 1983.
23. Schutz, A. and Luckmann, T. *Structures of the Life World.* Northwestern University Press, Evanston, Ill., 1973.
24. Sparck Jones, K. *Automatic Keyword Classification for Information Retrieval.* Archon, 1971.
25. Sparck Jones, K. Automatic indexing. *J. Doc. 30,* 4 (1974), 393–432.
26. Su, L.T. Evaluation measures for interactive information retrieval. *Inf. Process. Manage. 28,* 4 (1992), 503–516.
27. Sundheim, B. Ed. *Proceedings of the Third Message Understanding Evaluation and Conference.* Morgan Kaufmann, Los Altos, Calif., 1991.
28. Tong, R.M., Appelbaum, L.A. and Askman, V.N. A knowledge representation for conceptual information retrieval. *Int. J. Intell. Syst. 4,* 3 (1989), 259–283.
29. Turtle, H. and Croft, W.B. Efficient probabilistic inference for text retrieval. In *Proceedings RIAO 3* (1991), pp. 644–661.
30. Turtle, H.R. and Croft, W.B. Evaluation of an inference network-based retrieval model. *ACM Trans. Inf. Syst. 3* (1991), 187–222.
31. Turtle, H.R. and Croft, W.B. A comparison of text retrieval models. *Comput. J. 35,* 3 (1992), 279–290.
32. van Rijsbergen, C.J. *Information Retrieval.* Butterworths, 1979.
33. Willett, P. Recent trends in hierarchic document clustering: A critical review. *Inf. Process. Manage. 24,* 5 (1988), 577–598.

CR Categories and Subject Descriptors: H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Retrieval models, search process*

General Terms: Performance

Additional Key Words and Phrases: Information filtering, information retrieval

About the Authors

NICHOLAS J. BELKIN is a professor in the School of Communication, Information and Library Studies at Rutgers University, and vice-chair of ACM SIGIR. Current research interests include interaction in information retrieval systems, interface design for information retrieval systems, and evaluation of interactive information systems. **Author's Present Address:** School of Communication, Information and Library Studies, Rutgers University, 4 Huntington Street, Room 311, New Brunswick, NJ, 08903; email: belkin@zodiac.rutgers.edu

W. BRUCE CROFT is a professor and the director of the NSF Center for Research on Intelligent Information Retrieval at the University of Amherst, and past chair of ACM SIGIR. Current research interests include formal models of retrieval for complex text-based objects, text representation techniques, and the design and implementation of text retrieval systems. **Author's Present Address:** Department of Computer Science, University of Massachusetts, Amherst, MA 01003; email: croft@perth.cs.umass.edu

This work was supported in part by the Air Force Office of Scientific Research under contract 91-0324.