IMPLICATIONS OF BOOLEAN STRUCTURE FOR PROBABILISTIC RETRIEVAL

Abraham Bookstein
Graduate Library School
University of Chicago
1100 E. 57th St.
Chicago, IL 60637

1. Introduction

The purpose of an information retrieval system is to retrieve documents in response to a request. However, there are many strategies as to how this can be accomplished. Most popular are the Boolean systems attached to the commercial on-line database systems. Less conspicuous are a number of approaches being developed in university research laboratories and which are the object of this paper.

This paper is divided into two parts: The first will be a more or less non-technical overview of research in IR over the last ten to fifteen years. The second part, which is a bit more technical and a lot more speculative, will cover some interesting questions that are now being studied.

2. Theoretical Research in Information Retrieval

Research in IR today stems from two traditions:

1. Logical (Boolean) [1]
2. Combinatoric [2]

Both are based on a set of index records that represent a set of documents, and both try to respond to the demand of retrieving the best set of documents, given a request for information from a patron. The two approaches initially differed in:

i) How the documents are indexed (for example, the combinatoric approaches sometimes assigned numeric weights to index terms);

ii) The structure of the requests. Boolean: familiar combination of terms using AND, OR, and NOT connectives; Combinatoric: ignores structure -- the request is a set of terms, perhaps weighted;

iii) Matching function: how the request is used to retrieve documents.

These approaches have often been hostile to each other. The purpose of this paper is to review the ways of looking at IR, with particular attention to the less familiar combinatoric approaches; to indicate the strengths and weaknesses of each; and to mention some recent work that suggests, at least in principle, that these polar approaches may be combined -- or, in any case, more properly contrasted. In particular, I will emphasize the role of probability theory in current thinking about IR, and the possibility of probability theory forming a bridge between these approaches.

Combinatoric

Documents are represented by sets of index terms, maybe weighted; this representation can take the form of a vector:

$$D = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix},$$

where the w's are weights assigned to index terms. Usually these will be zero; sometimes they are restricted to taking values only of

This paper is based upon work supported, in part, by the National Science Foundation under Grant IST-8413559.

zero or one, to denote a term being assigned or not to a document.

Requests are similarly represented:

$$R= \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix}$$

Given this representation, the IRS selects documents by assessing the similarity between D and R, and retrieves those documents with the greatest similarity. Many matching functions are possible, for example, a simple count of overlapping terms.

Representing documents and requests by vectors suggests a geometric concept space, analogous to physical space. Thus measures of "distance" between R and D can be used; most popular have been measures related to the cosine of the angle between the two vectors.

Advantages

1. Simple: Don't need to construct a correct Boolean request, which seems to be difficult for many people, and intrinsically difficult for many requests [3].

2. The analogy to a physical geometric space is very satisfying.

3. Flexible: Neither users nor indexers need to be restricted to assigning or not assigning a term. An indexer, for example, who would be uncertain whether to assign a term in a conventional Boolean system can give it a weight of, say, .5 in a vector system.

   Similarly, users can distinguish between different degrees of importance in formulating their requests.

4. It permits a ranking of documents: instead of simply presenting a set of documents, it can also suggest an order in which the documents should be considered.

5. It permits a response to feedback - both for requests and index records. The idea is that there is a best place for the document or request to be in this concept space, and techniques exist that permit these vectors to migrate toward that ideal location: as we learn for which requests a document is successfully or unsuccessfully retrieved, we can modify its indexing, bringing it closer the to successful requests and farther from the unsuccessful requests [4].

Disadvantages

1. Can't represent the structure inherent in most requests: for example, "A AND B" and "A OR B" are both represented by the same vector and retrieve the same documents with the same weights.

2. Certain conceptually important aspects of IR are missing: in particular, role of uncertainty.

3. No theoretical foundation existed for the earliest algorithms, so *ad hoc* rules are used for assigning weights, computing similarities, and modifying weights in response to feedback.

This then was the state of IR as of about 1975: two apparently radically differing models, one popular in implementations, the other in the laboratory, with different features, different advantages, existing independently.

I chose 1975 as the dividing line because at about that time the first of the current generation of papers appeared using probability ideas to develop retrieval algorithms [5,6].

Probability is the body of mathematics developed for dealing with uncertainty. In IR, the probabilities are numbers measuring, given a document's index record (or other clues about the document), how likely is it that the document will be relevant to the request at hand. It permits us to make explicit the uncertainty intrinsic to IR.

Probability theory, as applied to IR, is based upon the following important ideas:

1) Probability: a number quantifying uncertainty.

2) Measures of costs: numbers assigning penalties to the

types of errors one can make, for example, retrieving bad documents (precision errors) or failing to retrieve good documents (recall errors).

3) Optimization: the prescription that one should act so as to minimize "expectea" cost, the "average" cost of acting in certain way.

A number of versions of this approach have appearea, differing in how uncertainty is represented, basically making different distributional assumptions regarding clues in relevant and non-relevant documents. But the underlying philosophy is very much the same: Retrieve next that document for which the expected cost is least. Also interesting is the importance of feedback in these approaches: to act well, it is important to know how likely it is that a relevant document will contain any index term. One starts with an intelligent guess -- and aids are provided to encourage intelligence. Afterwards, one makes use of users' assessments to improve on this estimate. For example, if out of 10 relevant documents retrieved, 8 have term A, we might estimate Prob (A|Rel) = .8 for subsequent calculations.

To bring us to about 1980, it is possible to say that the flavor of the resulting algorithms looked a lot more like the combinatoric algorithms than like the Boolean ones; the sequence of steps followed in a probabilistic IRS was as follows:

1. Assess initial probabilities;

2. Use these probabilities to assign weights to terms;

3. For each document, add weights corresponding to the terms associated with that document;

4. Retrieve highest weighted documents;

5. Reassess probabilities;

6. Go to step 2 or stop.

In this manner, the combinatoric approach suddenly found an unexpected ally in probability theory, which gave its procedures a theoretically coherent base, and filled in gaps on, for example, how to weight terms or use feedback.

About this time the notion of fuzzy-set theory was introduced, or at least gained some popularity in IR, and changed our view as to the possibilities available within the Boolean approach [7,8]. Without going into detail about the mechanics, I might note that fuzzy-sets:

1. Generalize the traditional sets of Boolean retrieval;

2. Allow one to preserve the logic of Boolean retrieval (use of AND, OR, NOT operators for set manipulation);

3. Assign weights to terms in documents; and

4. Rank retrieved documents.

Although the ranking is not very sensitive to the details of the documents' index records, the fact of ranking alleviates an important weakness for which Boolean systems have been condemned, and reduces one of the most important advantages claimed by combinatoric retrieval.

It might be useful to mention a fine point of some conceptual importance in bringing us up to date: the interpretation of the retrieval weights. To compare the meaning of weights in Boolean/fuzzy IR and in probabilistic IR, we observe:

Fuzzy sets: 1. Recognized no uncertainly; and

2. Weights directly represented degrees of aboutness; whereas

Probabilistic retrieval:

1. Emphasized uncertainty; and

2. Recognized only two levels of aboutness: about/not about (weights were probabilities of being in one or the other class).

Thus, though both methods might assign a value of .8 to a document, and this value could be used for locating that document on a ranked list, this value means very different things in the two systems. However, with the introduction of weights in Boolean systems, we see the beginnings of a convergence of the two approacnes. Furthermore, more recently [9], it has become possible to show that probabilistic models can be developed that do include intrinsic relevance classes, very much as fuzzy set retrieval does: the probabilities and costs that drive these models can refer to a document being in any of a number of relevance classes, not just "relevant" and "not relevant."

We can summarize the above, then, as follows:

Fuzzy set/Boolean

1. Can represent the fact that documents are about a request in differing degrees.
2. Weights cannot readily be assigned to terms in the request.
3. Can provide ranked output.
4. Doesn't easily exploit feedback.
5. Request structure critically important, and, in fact, defines this approach.
6. Intrinsically deterministic.

Probabilistic/combinatoric

1. Also can represent the fact that documents are about a request in differing degrees.
2. Also cannot readily assign weights to terms in a request. It is interesting to note that this actually represents a regression from the earlier vector models in which weights were assigned to terms both in the request and in the document. However, terms in the request become weighted as a consequence of feedback information as used in probabilistic retrieval algorithms. Ultimately, request weights do appear and reflect the distributional and discrimination value of the terms.
3. Also can provide ranked output -- and probably does so better.
4. Feedback information is critical to most probabilistic algorithms.
5. No structure, or only simple structure, is used.
6. Uncertainty central.

3. Speculations

We began with two radically different theoretic models of IR, and found that, over time, each has been able to appropriate some of the strengths of the other, at least in principle. At this point, the most striking conceptual differences separating these models are the deterministic character of the Boolean approach and lack of request structure in probabilistic systems. But are these differences inherent in the two approaches? My conjecture is, at least in principle, no, though, with a more complete theory, the details of how algorithms are derived will diverge from the older approaches, and implementations will be different. (See ref. [10] for an alternative approach toward merging Boolean and Probabilistic retrieval.)

To continue it will be useful to review the traditional derivation of the probabilistic algorithms, to see whether this can be modified to incorporate Boolean structure.

Using, for simplicity, a two value relevance scale, we retrieve a document if

$$c_1 \Pr(\bar{R} | x,r) < c_2 \Pr(R | x,r),$$

where
c_1 = cost of retrieving bad document
c_2 = cost of missing good document
r = request

With not very complex manipulation, now standard in IR, we conclude that we should retrieve a document if

$$\text{A)} \quad \frac{\Pr(R|x,r)}{\Pr(\bar{R}|x,r)} > \frac{c_1}{c_2} = \text{(threshold const)}$$

or

$$\text{B)} \quad \Pr(R|x,r) > \frac{c_1}{c_1 + c_2}$$

The second form states that we should retrieve documents in decreasing order of probability, a rule sometimes referred to as the "probability ranking principle" [11].

Usually probability manipulation is used to get the rule (from B): retrieve if

$$\frac{\Pr(x|R,r)}{\Pr(x|\bar{R},r)} > \text{Const.,or} \sum \log \frac{[p/1-p]}{[\bar{p}/1-\bar{p}]} = \sum w_i$$

the sum being taken over the terms appearing in the document. This derivation assumes $P(x) = P(x_1)P(x_2) \ldots P(x_n)$, the term independence assumption; the p's and $\bar{p}$'s are the probabilities of a term occurring in relevant and non-relevant documents.

Recently a great deal of attention has been given to overcoming the limiting term independence assumption. Usually this has involved rather complicated, unsatisfying and not very effective expansions of the distribution function. Perhaps we can save some of the simplicity of the term independence models and at same time introduce some Boolean structure. Let me at this point suggest that the term independence model effectively assumes a Boolean request of the form A AND B AND C, in the sense that introducing probability assumptions into a simple Boolean model restricted to AND connectives would yield a form very similar to that described above. But all Boolean requests could be reformulated as:

a) $(t_1$ OR t_2 OR ...) AND $(t_i$ OR ..) AND ...

which looks like the simple conjunction of a number of terms. Thus, our knowledge of how to treat the AND connective within a probabilistic framework could be used to process a general Boolean request if we knew how to treat the OR connective; for example, we can treat components such as t_1 OR t_2 OR ... as a single term (or "hyperterm" [12]) and, with respect to these hyperterms, use the conventional term independence model.

In the proposed model, t_1, t_2, etc., are treated in the analysis as if they were a single term. Suppose, however, that both t_1 and t_2 occur. We can throw this information out, or else model multiple occurrences. For example, to borrow a model used successfully elsewhere, we can estimate the probability of relevance, given that some of the t's making up a hyperterm occur, by the expression:

$$\frac{e^{(a_1 t_1 + a_2 t_2 \,..\, + a_n t_n)}}{1 + e^{(\quad)}}$$

The a's are parameters, to be estimated, for example, by means of maximum likelihood techniques, from feedback data. This model has the advantage of not only recognizing the occurrences of more than one component of the hyperterm, but also of being able to give different weights to each component.

Thus we see it may be possible to introduce structure in a probabilistic model. Can we introduce uncertainty into a Boolean model?

In the simplest models we rank documents by Pr{R|x,r}. In the usual developments, r (and thus R) is considered as a unit, without structure, and the effort is to interchange the roles of R and x, so that the structure of x can be exploited. But if r has structure, in the Boolean sense, so would R. For example, if

I. r = a AND b, then R would mean A and B, that is, that the document is relevant to both A and B

Keeping the same assumptions as traditionally made:

Pr{A and B|x} = Pr{A|x)Pr{B|x}.

If we assume A influences mainly x_a, an assumption consistent with the independence assumptions commonly made, we straightforwardly get as a weight function:

$$\frac{\Pr\{x_a|A\}}{P\{x_a|\bar{A}\}} \, \frac{\Pr\{x_b|B\}}{P\{x_b|\bar{B}\}} \text{, which is equivalent}$$

to the weight function

$$\sum_{AB} \log \frac{p/(1-p)}{\bar{p}/(1-\bar{p})},$$

a result very similar to the results of earlier derivations that ignore Boolean structure.

So, in form, the earlier probability models seem to be assuming a request of the form a AND b AND c ..., for independent concepts.

II. r = a OR b

Two extreme models suggest themselves:

a) Synonym model: a, b are synonyms for concept A, and the indexer chooses one or the other, occasionally both.

$$\frac{\Pr\{R|x\}}{\Pr\{\bar{R}|x\}} = \frac{\Pr\{x_a \text{ or } x_b|A\}}{\Pr\{x_a \text{ or } x_b|\bar{A}\}}$$, which in effect reduces to the model already described, if x_a and x_b are considered as terms making up a hyperterm.

b) Independence model: A and B are independent concepts.

Now, $\Pr\{A \text{ or } B|x\} = \Pr\{A|x\} + \Pr\{B|x\} - \Pr\{A|x\}\Pr\{B|x\}$

More generally, if we can assume concept independence,
$\Pr\{A \text{ and } \ldots \text{ and } Z\} = P_A \ldots P_Z$ and
$\Pr\{A \text{ or } \ldots \text{ or } Z\} = 1 - (1 - P_A) \ldots (1 - P_Z)$
would form the basis of retrieval; here
P_A denotes $\Pr\{A|x\}$, etc.

4. Conclusions

Although the above derivations were sketched out for a simple two value relevance scale, similar, though more complex, formulae would apply if a special fuzzy-set model were used. Instead of simply being able to indicate whether a document was relevant or not to a concept (P_A and $1 - P_A$ above), we would now need to use a probability distribution over the possible relevance classes. It is interesting to speculate whether such an approach might lead to an acceptable way of assigning request weights within a fuzzy set system, a task whose solution has been hitherto very elusive. A number of very serious technical problems remain, however, before such an approach can be tested. For example, the problem of how to treat term dependencies is as nettlesome as ever. But more serious, the tasks of probability estimation and how to take advantage of feedback information remain to be solved.

However, our concluding speculation seems particularly tempting, given the above development: That the usual algorithms used in probabilistic retrieval are in effect the result of a Boolean system, where uncertainty is introduced and the requests are restricted to a particularly simple form. Thus the claim that probabilistic retrieval is intrinsically simpler than Boolean retrieval may not be completely accurate. Rather, the simplicity may result from in fact making simplifications within a Boolean framework. Put this way, of course, the relative simplicity of the traditional probabilistic approaches is less impressive.

References

1. Lancaster, F.W. Information Retrieval Systems: Characteristics, Testing and Evaluation (2nd Ed). New York: Wiley-Interscience. 1979.

2. Salton, G. and McGill, M.J. Introduction to Modern Information Retrieval. New York: McGraw Hill. 1983.

3. Cooper, W.S. "Exploiting the Maximum Entropy Principle to Increase Retrieval Effectiveness." Journal of the American Society for Information Science. 1983; 34 (1): 31-39.

4. Rocchio, J.J. Relevance Feedback in Information Retrieval. In: Salton, G. (ed.) The SMART Retrieval System: Experiments in Automatic Document Processing. Englewood Cliffs, N.J.: Prentice-Hall. 1971.

5. Bookstein, A. and Swanson D. "A Decision Theoretic Foundation for Indexing." Journal of the American Society for Information Science. 1975; 26: 45-50.

6. Robertson, S.E. and Sparck-Jones, K. "Relevance Weighting of Search Terms." Journal of the American Society for Information Science. 1976; 27: 129-146.

7. Radecki, T. "Fuzzy Set Theoretical Approach to Document Retrieval." Information Processing and Management. 1979; 15: 247-59.

8. Bookstein, A. "Fuzzy Requests." Journal of the American Society for Information Science. 1980; 31 (3): 240-47.

9. Bookstein, A. "Outline of a General Probabilistic Retrieval Model." Journal of Documentation. 1983; 39 (2): 63-72.

10. Salton, G., Fox, E.A. and Wu, H. "Extended Boolean Information Retrieval." Communications of the ACM. 1983; 26 (12): 1022-1036.

11. Robertson, S.E. "The Probability Ranking Principle in IR." Journal of Documentation. 1977; 33: 294-304.

12. Tague, J. and Nelson, M. "Simulation of Bibliographic Databases Using Hyperterms." In: Salton, G. and Schneider, H-J (eds). Research and Development in Information Retrieval. Berlin: Springer-Verlag; 1983.