AUTOMATIC ASSIGNMENT OF SOFT BOOLEAN OPERATORS

Gerard Salton and Ellen Voorhees*
Department of Computer Science
Cornell University
Ithaca, NY 14853 USA

Abstract

The conventional bibliographic retrieval systems are based on Boolean query formulations and inverted file implementations. Such systems provide rapid responses in answer to search queries but they are not easy to use by uninitiated patrons. An extended Boolean retrieval strategy has been devised in which the Boolean operators are treated more or less strictly, depending on the setting of a special parameter, known as the p-value. The extended system is much more forgiving than the conventional system, and provides better retrieval effectiveness. In this study various problems associated with the determination of appropriate p-values are discussed, and suggestions are made for an automatic assignment of p-values. Evaluation output is included to illustrate the operations of the suggested procedures.

1. Introduction

In most operational information retrieval systems, Boolean logic is used as a controlling factor in distinguishing the items that are to be retrieved from those that must be rejected. It has been remarked that the conventional Boolean logic exhibits many shortcomings in a retrieval setting: [1,2]

a) The conventional Boolean logic is strict and unforgiving. Thus, adding an **anded** term may turn a viable query statement into an excessively narrow formulation with correspondingly poor recall; analogously, adding an **ored** term may excessively broaden the query and produce correspondingly poor precision.*

b) The amount of output produced by a Boolean query is difficult to control even for trained experts who are familiar with the vocabulary use in the collection under consideration. Too often a given formulation produces too little or too much output to be of real use.

c) The output obtained from conventional Boolean systems is not ranked in any order of presumed usefulness for the user. This makes it impossible for the user to consider the more important items ahead of the more marginal ones.

d) In conventional Boolean logic, term importance weights are not normally used to distinguish the relative importance of terms assigned to the documents or included in the queries. That is, a given term is assumed to be fully present, or fully absent from a given description.

e) In conventional systems it is not possible to "oversatisfy" a query--for example, in response to query (A **or** B) a document containing both terms is not preferred over one containing only one term. Similarly, it is not possible to "undersatisfy" a query--for example, in response to query (A **and** B **and** C), a document containing two of the three terms is rejected as completely as a document containing no terms at all.

An alternative to the conventional Boolean retrieval system is the vector processing system where queries are formulated as sets of terms without Boolean operators. [3,4] In the vector processing system it is easy to operate with

This study was supported in part by the National Science Foundation under grant IST 83-16166.

© 1985 ACM 0-89791-159-8/85/006/0054 $00.75

*Recall is the proportion of relevant items retrieved, whereas precision is the proportion of retrieved items that are relevant.

weighted query and document terms, and a global similarity value can be computed between a query and the individual documents. The output can then be controlled and the documents can be ranked for retrieval in decreasing order of the query-document similarity.

The extended Boolean retrieval method is based on a strategy similar to that used in vector processing except that Boolean, or quasi-Boolean, queries are used. [5] In particular, a global similarity measure is computed between a Boolean query and the documents identified by sets of weighted terms. This makes it possible to arrange the documents at retrieval time in decreasing order of the corresponding query-document similarity. In the extended Boolean model, both query and document terms may be weighted and the interpretation of the Boolean operators is controlled by a special parameter, known as the p-value.

Consider, in particular, a document $D = (d_A, d_B, d_C, \ldots)$, where d_i specifies the importance weight of document term $i, 0 \leq d_i \leq 1$. Let a,b,c, etc. represent the weights of *query* terms A, B, C and so on. Then **or**- and **and**-queries may be defined respectively as

$$Q_{\mathbf{or}} = [(A,a)\ \mathbf{or}\ (B,b)\ \mathbf{or}\ (C,c)\ \mathbf{or} \ldots] \tag{1}$$

and $Q_{\mathbf{and}} = [(A,a)\ \mathbf{and}\ (B,b)\ \mathbf{and}\ (C,c)\ \mathbf{and} \ldots]$.

The following values may then be used to denote the similarity between D and $Q_{\mathbf{or}}$, and D and $Q_{\mathbf{and}}$ respectively:

$$\mathrm{sim}(D,Q_{\mathbf{or}}) = \left[\frac{a^p d_A{}^p + b^p d_B{}^p + c^p d_C{}^p + \ldots}{a^p + b^p + c^p + \ldots}\right]^{1/p}$$

$$\mathrm{sim}(D,Q_{\mathbf{and}}) = \tag{2}$$

$$1 - \left[\frac{a^p(1-d_A)^p + b^p(1-d_B)^p + c^p(1-d_C)^p + \ldots}{a^p + b^p + c^p + \ldots}\right]^{1/p}$$

For mixed queries with both **and** and **or** operators, the **or** and **and** formulas of expressions (2) are appropriately combined. For example, for the query $Q = \{ (A,a)\ \mathbf{or}\ ([(E,e)\ \mathbf{and}\ (F,f)] ,b) \}$ one obtains

$$\mathrm{sim}(D,Q) = \tag{3}$$

$$\left\{\frac{a^p d_A{}^p + b^p \left[1 - \left(\left\{\frac{e^p(1-d_E)^p + f^p(1-d_F)^p}{e^p + f^p}\right\}\right)^{1/p}\right]^p}{a^p + b^p}\right\}^{1/p}$$

It is not difficult to show that all similarity formulas produce values between 0 and 1. Furthermore the value of the parameter p determines the strictness of interpretation of Boolean **and** and **or** operators: [5,6]

a) When $p = \infty$, and binary weights are used, the result is a conventional Boolean system. The similarity measurements provide values of 1 for documents that match the queries and hence are to be retrieved, and values of 0 for the remainder that needs to be rejected.

b) As the value of p is reduced from infinity, the interpretation of the Boolean operators is less and less strict. A clause such as (A **and** B) is then treated as a tentative phrase, rather than a compulsory one, in the sense that a document exhibiting both terms still receives a perfect score; however a document exhibiting one of the two terms now receives a partial sum rather than a null score as in the conventional system. Analogously, in a clause such as (A **or** B), the two terms are treated as approximately, rather than completely, synonymous. This implies that a document which includes both terms will be preferred over one containing only one of the terms.

c) As p reaches its lower boundary of 1, the two formulas of expression (2) produce the same result; that is $\mathrm{sim}(D,Q_{\mathbf{or}}) = \mathrm{sim}(D,Q_{\mathbf{and}})$. In that case, the presence of two query terms in a document is worth twice as much as the presence of one term (assuming fully weighted terms), and the distinctions between **and** and **or** are lost. When $p = 1$ the extended Boolean system is reduced to a vector processing system, and the Boolean queries (A **and** B) and (A **or** B) are then equivalent to the vector query (A,B) specifying two terms without special structural information.

In order to use the extended Boolean system, it is necessary to specify appropriate weights to be attached to query and document terms, and to choose the p-values controlling the interpretation of the operators. It is known that a high order of performance is obtained when high weights are used for terms with high occurrence frequencies in individual documents but low overall occurrence frequencies in the remainder of the collection. For this reason an appropriate weight for term j in document i might be

$$w_{ij} = \frac{\text{occurrence frequency of term j in document i}}{\text{total number of documents exhibiting term j}}. \tag{4}$$

This weighting system also known as tf × idf (term frequency × inverse document frequency) has a term significance component (the term frequency tf), and a term relevance component (the inverse document frequency idf). The inverse document frequency is known to be useful as an approximation to the probabilistic term relevance. [7,8]

For query terms, the occurrence frequency of each individual term is assumed to be equal to 1, and weights are then computed recursively for the query clauses. The denominator of expression (4), representing the document frequency is used for both query and document term weighting.*

2. Overall Effect of the Extended Boolean Operations

It is known that the retrieval effectiveness of the extended, relaxed Boolean system is far superior to that of the conventional Boolean logic. When searches are evaluated in terms of recall and precision, a single search iteration in a properly chosen extended system may produce improvements in the average precision computed for various recall points ranging from 50 to 100 percent over the average precision provided by the conventional logic. [5] In an iterative search based on relevance feedback several search steps are normally carried out with progressively improved query statements. In that case, the improvement of the extended over the conventional system may reach several hundred percent after three or four search iterations, depending on the collection and the queries under consideration. [9]

The basic reason for the improved performance is the forgiving nature of the relaxed Boolean system. In the extended system, the appropriateness of a particular Boolean **and** or **or** operator is much less critical than in the conventional Boolean system and the query formulations are more differentiated. Two main performance differences may be noted between conventional and extended systems:

a) Many relevant documents cannot be retrieved using the conventional logic because the specified query clauses are not satisfied by the document term sets. As will be seen, many of these items are nevertheless retrievable early in a search in the extended system.

b) Many relevant documents that are actually retrieved in a conventional Boolean system may be retrieved in the extended system with considerably improved retrieval ranks.

Consider as an example of case (a) document 1207 of the ISI collection and its performance in response to query 16 as illustrated in Table 1*. The natural language query appears at the top of Table 1 followed by the Boolean formulation. The superscripts attached to the Boolean operators represent manually chosen p-values about which more will be said later in this note. Excerpts from the document title and abstract appear below the query information in Table 1. It is clear from the query and document texts that the Boolean query formulation is not satisfied by the excerpted document text. In particular, the first **anded** query clause is not present in the document since the term "retrieval" does not occur in either title and abstract. This accounts for the fact that when p $= \infty$, the document is not retrievable, the corresponding query-document similarity being equal to 0. The retrieval rank of 917 out of 1460 documents shown in Table 1 for the Boolean search is a simulated rank produced by the system. In the simulated ranking operation all retrieved documents with a query-document similarity of 1 will of course exhibit lower ranks than the rejected items with a 0 correlation.

Even though document 1207 is not retrievable by a conventional Boolean search, the document is the second highest retrieved by a search in which weighted terms are used and uniform p-values of 2 are attached to all operators. When uniform p-values of 1 are used, the document is retrieved with a rank of 3. When the mixed p-values shown in the query formulation of Table 1 are used, the

*The following normalized weighting formulas are used in practice to obtain the weights of term j in document i (or query i) respectively.

a) document terms: $w_{ij} =$

$$\left\{ \left[0.5 + 0.5 \left(\frac{\text{frequency of term j in doc i}}{\text{max. freq. of any term in doc i}} \right) \right] \times \left[\ln \left(\frac{\text{number of docs in collection}}{\text{number of docs with term j}} \right) \right] \right\}$$

b) query terms: $w_{ij} =$

$$\ln \left(\frac{\text{number of docs in collection}}{\text{number of docs with term j}} \right) \div \ln \left(\frac{\text{number of docs in collection}}{1} \right)$$

c) query clause: average term weight of the terms in a clause.

*All examples included in this study are based on retrieval operations performed with a collection of 1460 documents in automatic documentation (the ISI collection) used with a set of 35 search requests.

retrieval rank is also equal to 2. The excellent performance of the extended Boolean system in this case is due to the near-match between query and document terms: a complete match exists with the second main query clause because "remote consoles" is present in the document text; in addition one finds two instances of the query term "information" in the document. The individual term weights, and hence the query-document similarity, will increase with increasing occurrence frequencies of matching terms. A query-document match with a highly occurring term is thus more beneficial than a match with a rare term. This accounts for the fact that a large number of documents that are not retrievable in a conventional Boolean system are in fact obtainable comparatively early in a search in the extended system.

Case (b) is illustrated by document 783 and ISI query 1 in Table 2. Here the document contains the query terms "title", "accuracy", and "information", and the Boolean clause "title **and** accuracy **and** information" in fact suffices for retrieval in the conventional environment. However, whereas document 783 is retrieved with a rank of 34 in the conventional system--there are 33 other documents with a perfect query-document similarity of 1 that happen to be retrieved earlier in the search--the retrieval ranks vary between 4 and 14 for the three lower p-values assignments included in the Table. The reason for the improved performance of the extended system may be found in the multiple matches of the several query terms listed in the lower half of Table 2.

Thus, in addition to making it possible to retrieve many relevant document not obtainable by conventional means, the extended system also lowers the retrieval ranks of many items that are also obtained by standard techniques.

3. Conventional Boolean versus Fuzzy Set Approaches

The fuzzy set retrieval model allows the assignment of variable weights to the terms attached to the documents of a collection, while providing compatibility with ordinary Boolean operations when the document term weights are restricted to 0 and 1. [10,11] In fuzzy set theory, the following measurements are used to evaluate the similarity between a document $D = (d_A,d_B)$ and the Boolean queries (A **or** B), (A **and** B), and (**not** A), respectively:

$$\text{sim}\ (\ D,\ A\ \textbf{or}\ B) = \max\ (d_A,\ d_B)$$

$$\text{sim}\ (\ D,\ A\ \textbf{and}\ B) = \min\ (d_A,\ d_B)$$

$$\text{sim}\ (\ D,\ \textbf{not}\ B) = 1 - d_A$$

When the term weights in the document are limited to 0 and 1 as in the conventional logic, the usual results are obtained because a retrievable document for **or** must exhibit a maximum term weight of 1, whereas a retrievable item for **and** has a minimum term weight of 1.

The fuzzy set approach can be simulated in the extended Boolean model by using ordinary Boolean queries with p-values set equal to infinity, but allowing variable weights for all document terms. Since the only effective difference between the pure Boolean and the fuzzy set approaches resides in the assignment of variable document term weights, a document which does not match a particular Boolean query, and hence receives a zero query-document similarity, will still receive a zero similarity coefficient in the fuzzy set system. Thus items that are not retrievable in a Boolean system are also rejected in the fuzzy set environment.

However, in the fuzzy set model the variable document weights can be used to assign to each retrieved item a variable query-document similarity, which is used in turn to rank the retrieved items for output. In many cases the fuzzy set system provides much better discrimination for the retrieved items than the conventional Boolean system. Consider, for example, ISI query 3 which retrieves 28 relevant documents out of a total of 42 relevant ones in the collection. The output of Table 3 shows that in the conventional Boolean system the relevant items appear between ranks 3 and 244; but only 8 relevant items are retrieved in the top 100 ranks. In the fuzzy set approach, the retrieval ranks for the relevant items vary between ranks 1 and 205, and 23 items appear in the top 100. The summary output provided for four document collections later in this study shows that the document term weights provide average improvements in recall-precision of at least 10 percent over the conventional Boolean model. For some document collections, the performance improvements can be much larger.

4. Extended Boolean System - Uniform p-Value Assignment

In the fuzzy set system only the document terms can be weighted. The query terms remain unweighted. When lower p-values are introduced the interpretation of the Boolean operators is relaxed, and term weights are then usable for document as well as query terms. In general one would expect that relatively higher p-value assignments implying stricter query structures would favor high precision output, whereas the lower p-values closer to 1 that approach the single term vector processing system would favor higher recall.

Furthermore, judging by the performance of the fuzzy set retrieval model, the system should perform better with weighted than with binary terms.

An overview of the performance of the extended Boolean system is presented in Table 4 for four collections of documents with corresponding query sets in computer science (the CACM collection comprising 3204 documents and 52 queries), documentation (the CISI collection with 1460 documents and 35 queries), electrical engineering (the Inspec collection with 12684 documents and 77 queries) and biomedicine (the Medlars collection comprising 1033 documents and 30 queries). [12] The output of Table 4 uses a single performance figure representing the average search precision evaluated at 3 recall points (recall of 0.25, 0.50, and 0.75), averaged over the complete query set for each collection.

The benchmark run is the conventional Boolean system included at the top of Table 4. This is followed by the performance figures for the fuzzy set model and for four uniform p-value assignments including $p = 2$, 1.5, 1, and 0.5. In each of the last four cases, all Boolean operators included in the query were set to a common, specified p-value, and weighted terms were used for both queries and documents. The extended Boolean model does not account for p-value assignments less than 1. When the p-values are reduced below 1, the functions of **and** and **or** become interchanged, and the meaning of the queries is substantially altered. As a strictly formal exercise, it is however possible to work with p-value assignments outside the range $[1,\infty]$.

The improvement of the fuzzy set model over the conventional Boolean system ranges from about 10 percent for CISI to nearly 46 percent in average precision for Inspec. Much more substantial gains are obtained when the p-values are lowered. For all collections the best average performance is obtained for a uniform p-value equal to 1. In fact, for three of the four collections one obtains $p(1) > p(1.5) > p(2) > p(0.5)$ as an overall result. In one case (for Medlars), the performance for $p = 0.5$ exceeds that for $p = 2$, but even there $p = 1.5$ and $p = 1$ are superior. The conclusion is that the Boolean operators ought to be relaxed as much as possible when uniform p-values are used, but that it is important to stay within the boundaries of the model (that is, to avoid p-values smaller than 1). For $p = 1$ the improvements for a single search operation ranges from 62 percent for CISI to over 177 percent for Medlars.

The results for the individual queries, as opposed to overall averages, are much more variable: sometimes $p = 1$ is preferred over $p = 2$, and sometimes the reverse is true, with $p = 1.5$ taking on intermediate values. Results for three particular query-document pairs are included as illustrations in Table 5. In each case, the document output ranks are shown for various p-value settings, and for query Q5 the query-document similarities are also included in the Table. For query Q1 and document 783, the preferred p-value is 2 since the document is then retrieved in fourth place. The retrieval ranks go up as the p-values decrease to 1 or increase to ∞. For query Q7 and document 725, $p = 1$ is preferred with a retrieval rank of 15; the retrieval ranks deteriorate as the p-values increase.

The right-most example in Table 5 shows that the size of the query-document similarity may point to a preferred parameter value even when equivalent retrieval ranks are obtained for various p-value settings. In this case, the query requested information about "training or instruction in information management and information retrieval", and the document text contained 6 matches with the term "instruction", and one additional match with "training". These single term coincidences led to an additional match with the query clause "training **or** instruction". The clause match accounts for the improved query-document similarity for $p = 2$ over that for $p = 1$ when matching clauses are not given any extra weights. It may be noted that document 1246 is not retrievable by a conventional Boolean search (0 correlation and simulated rank of 226) even though it is retrieved at the very top in the extended system. In that case, the Boolean formulation included additional **and**ed terms, such as "retrieval", which were not satisfied by the document.

5. Extended Boolean System - Mixed p-Value Assignments

In the experiments reported in the previous section, the same p-values were assigned uniformly to all Boolean operators. One may ask whether better retrieval effectiveness may be achieved by differentiating among the p-values depending on the strength of connection of the individual terms in a given Boolean clause. The following basic strategy suggests itself:

a) When an **and**ed term pair constitutes a standard noun phrase in the language, or when the semantic relationship between the components is strong, the corresponding p-value should be large.

b) When the terms in an **or**ed clause are strict synonyms, that is, when the components are usable interchangeably in the same context, the corresponding p-value should be large.

c) When these conditions are not met and the term relationships are weak, the p-values should be close to 1.

An examination of the available Boolean query formulations reveals that the components related by **and** operators often relate acceptable English phrases; on the other hand, the elements linked by **or** operators appear to be only vaguely related. This suggests that it may be useful to assign relatively low p-values to the **or**-operators, but higher p-values to those **and** operators that relate acceptable English phrases.

Such a strategy was used for a manual assignment of mixed p-values to the queries in the CACM and CISI collections. Thus the original query (ISI Q3)

(information **and** (science **or** definition))

was processed as

(information $\textbf{and}^2$ (science $\textbf{or}^{1.5}$ definition))

using the assumption that "information science" is in fact a reasonable phrase, whereas "science" and "definition" are not close synonyms. The sample output of Table 6 shows that the resulting queries do not generally outperform the best uniform automatic p-value settings. However in most cases, the mixed parameter assignment approximates the output obtained with the best uniform choice of the p-values.

For query ISI Q9 and document 1129, the best performance is obtained for $p = 2$, and the manual assignment of mixed p-values is almost as effective as the uniform assignment of $p = 2$. For query ISI Q17 and document 376, the best performance is produced by a uniform assignment of $p = 1$, and the mixed p-values are again nearly as effective. Finally for query ISI Q4 and document 420, the best performance with a retrieval rank of 20 is actually obtained with the manual mixed p-values. It may be noted that all of the documents displayed in Table 6 are rejected using a conventional Boolean approach.

The relative success of the mixed manual p-value assignment suggests that the procedure could be mechanized by using different, but automatically determined p-values for **and** and **or** operators. The previously given argument about the relative strictness of **and**ed and **or**ed clauses further suggests that relatively higher p-values be used for **and** than for **or**. Four different combinations of mixed p-values were used experimentally with the four sample collections, ranging from $p_{\textbf{and}} = 3.5$, $p_{\textbf{or}} = 1.2$ to $p_{\textbf{and}} = 2.0$, $p_{\textbf{or}} = 1.2$. The summary output of Table 7 shows that the best performance is obtained for $p_{\textbf{and}} = 2.5$, $p_{\textbf{or}} = 1$, with an alternative $p_{\textbf{and}} = 2$, $p_{\textbf{or}} = 1.2$ being second best. The manual mixed p-value assignments for CACM and CISI included at the bottom of Table 7 provide performance levels comparable to the second best automatic mixed assignments. The output of Table 7 also shows that the mixed p-values somewhat outperform the best uniform assignment of $p = 1$.

Complete recall-precision tables for the four query and document collections are presented in Table 8. The search precision is given in each case for 10 different recall values averaged over each particular query set. The best output performance is highlighted by double bars in the display of Table 8. It is seen that the preferred p-value setting is the mixed assignment of $p_{\textbf{and}} = 2.5$, $p_{\textbf{or}} = 1$ with a uniform assignment of $p = 1$ being a close second. The improvement of the mixed p-value assignment over the conventional Boolean search ranges from 55 percent for CISI to 155 percent for Inspec.

6. Extended Boolean System - Reformulated Queries

The Boolean query sets used with the four experimental collections contained a number of short formulations with very few terms. Such formulations are often preferred in conventional Boolean environments where a large number of clauses and operators may produce uncertain results. In the extended Boolean system, longer, more complete query formulations may become more productive. An attempt was therefore made to rephrase some of the shorter queries for CACM and CISI, and to render them more precise. The reformulation was carried out manually (intellectually) as shown by the example of Table 9.

The examples of Table 9 show that the reformulated queries are more complex than the original ones and contain a larger number of phrases; furthermore mixed p-values are also assigned as shown in the display. The summary output of Table 10 contrasting the performance of original versus reformulated queries shows that in the conventional Boolean mode, the new queries barely outperform the original ones for CACM; for CISI the new queries are actually less effective than the original ones. In the extended system, on the other hand, the new queries outperform the original ones for both the uniform as well as the mixed p-value assignments. The reformulated queries are about ten percent better than the original ones.

The experiments described in this study confirm that the extended Boolean system is useful and effective. The output of Table 8 provides performance data for the various p-value assignments.

References

[1] W.S. Cooper, (1982) The maximum entropy principle and its application to the design of probabilistic retrieval systems. Information Technology: Research and Development, 1:2, 99-112.

[2] A Bookstein, (1978) On the perils of merging boolean and weighted retrieval systems. Journal of the ASIS, 29:3, May, 156-158.

[3] G. Salton, ed., (1971) The Smart retrieval system: experiments in automatic document processing. Prentice Hall Inc., Englewood Cliffs, NJ.

[4] G. Salton and M.J. McGill, (1983) Introduction to modern information retrieval. McGraw Hill Book Company, New York.

[5] G. Salton, E.A. Fox and H. Wu, (1983) Extended boolean information retrieval. Communications of the ACM, 26:11, November, 1022-1036.

[6] H. Wu, (1981) On query formulation in information retrieval. Doctoral Thesis, Department of Computer Science, Cornell University, Ithaca, NY, January.

[7] W.B. Croft and D.J. Harper, (1979) Using probabilistic models of document retrieval without relevance information. Journal of Documentation, 35, 285-295.

[8] H. Wu and G. Salton, (1981) A comparison of search term weighting: term relevance vs. inverse document frequency. SIGIR Forum, 16:1, Summer, 30-39.

[9] G. Salton, E.A. Fox and E. Voorhees, (1983) Advanced feedback methods in information retrieval. Technical Report 83-570, Department of Computer Science, Cornell University, Ithaca, NY, August.

[10] T. Radecki, (1976) Mathematical model of information retrieval based on the concept of a fuzzy thesaurus. Information Processing and Management, 12:5, 313-318.

[11] A. Bookstein, (1980) Fuzzy requests: An approach to weighted boolean searches. Journal of the ASIS, 31:4, July, 240-247.

[12] E.A. Fox, (1983) Characterization of two new experimental collections in computer and information science containing textual and bibliographic concepts. Technical Report 83-561, Department of Computer Science, Cornell University, Ithaca, NY, September.

Query ISI Q16	
natural language:	systems incorporating multiprogramming or remote stations in information retrieval
Boolean form (partial statement):	{[(information **and** ^{2} retrieval) **or** $^{1.5}$ (....)] **and** $^{1.5}$ [(multiprogram **or** $^{1.7}$ (remote **and** $^{1.5}$ terminal) **or** $^{1.7}$ (remote **and** $^{1.5}$ console) **or** $^{1.7}$ (....)]}
Document 1207	
Title:	Technical *Information* Project
Abstract:	.... technical *information* system ... uses *remote consoles* ...

Retrieval Results	Retrieval Rank	Query-Document Similarity
$p = \infty$ Boolean	917	0.0
$p = \infty$ weighted	917	0.0
$p = 2$	2	.0968
$p = 1$	3	.0604
p mixed	2	.0796

Query-Document Matches

a) information (2 occurrences)

b) (remote **and** console) (1 occurrence)

Example for Document Not Retrievable by Conventional Boolean Search

Table 1

Query ISI Q1	
natural language:	what difficulties arise in retrieving articles from approximate titles
Boolean form (partial statement):	{title **and** ^{2} [(descriptive **or** $^{1.2}$ content **or** $^{1.2}$ approximate **or** $^{1.2}$ accuracy **or** $^{1.2}$ meaning ...) **and** ^{2} (retrieval **or** $^{1.2}$ information) ....]}
Document 783	
Title:	Author versus *Title*: A Comparative Survey of the *Accuracy* of the *Information*
Abstract:	.... comparative *accuracy* of author and *title information* .... show the *title* to be more *accurate*

Retrieval Results	Retrieval Rank	Query-Document Similarity
p = ∞ Boolean	34	1.0
p = ∞ weighted	10	.0802
p = 2	4	.0676
p = 1	14	.0477
p mixed	7	.0495

Query-Document Matches

a) title (3 occurrences)
b) accuracy (3 occurrences)
c) information (2 occurrences)
d) (title **and** ^{2} accuracy **and** ^{2} information)

Example for Document Retrievable by Conventional Boolean Search

Table 2

Query ISI Q3	Ranks of the Relevant Items	Number of Relevant Items Retrieved in the Top 100
pure Boolean ($p = \infty$, unweighted)	$3 \leq r \leq 244$	8
fuzzy set model ($p = \infty$, weighted)	$1 \leq r \leq 205$	23

Sample Comparison Between Boolean and Fuzzy Set Models

Table 3

		CACM 3204 52 queries	CISI 1460 35 queries	Inspec 12684 77 queries	Medlars 1033 30 queries
pure Boolean		.1512	.1037	.0998	.1943
$p = \infty$, fuzzy set (weighted docs.)		.1720 (+ 13.8)	.1147 (+ 10.7)	.1456 (+ 45.9)	.2191 (+ 12.8)
$p = 2$		.2558 (+ 69.2)	.1676 (+ 61.6)	.2517 (+ 152.2)	.5129 (+ 164.0)
$p = 1.5$	weighted document and query terms	.2585 (+ 71)	.1690 (+ 63.1)	.2543 (+ 154.8)	.5360 (+ 175.9)
$p = 1$		.2594 (+ 71.6)	.1681 (+ 62.1)	.2558 (+ 156.2)	.5386 (+ 177.3)
$p = 0.5$		.2451 (+ 62.2)	.1628 (+ 57)	.2258 (+ 126.2)	.5274 (+ 171.5)

Evaluation Summary for Uniform p-Value Assignment

(Precision Values for Recall of 0.25, 0.50, 0.75 averaged over
a Query Set for Each Collection)

Table 4

	ISI Q1 Document 783	ISI Q7 Document 725	ISI Q5 Document 1246	
p = 1	14	15	1	.0943
p = 1.5	6	21	1	.1344
p = 2	4	27	1	.1658
p = ∞ weighted	10	19	226	0
p = ∞ Boolean	34	456	226	0

Output Ranks for Three Specified Relevant Documents
in Response to Queries (uniform p)

Table 5

	ISI Q9 Document 1129		ISI Q17 Document 376		ISI Q4 Document 420	
	Rank	Similarity	Rank	Similarity	Rank	Similarity
p = 1	95	.0216	6	.0390	58	.0174
p = 1.5	10	.0431	10	.0374	42	.0219
p = 2	6	.0643	28	.0351	46	.0243
p mixed (manual)	8	.0510	10	.0379	20	.0216
p = ∞ (Boolean)	150	0	1348	0	802	0

Output Ranks and Query-Document Similarities
for Three Document-Query Pairs (mixed p)

Table 6

	CACM 3204 52 queries	CISI 1460 35 queries	Inspec 12684 77 queries	Medlars 1033 30 queries
pure Boolean	.1512	.1037	.0998	.1943
p = 1	.2594 (+ 71.6)	.1681 (+ 62.1)	.2558 (+ 156.2)	.5386 (+ 177.3)
p **and** = 2, p **or** = 1.2	.2595 (71.7)	.1701 (64.1)	.2579 (158.4)	.5401 (178.0)
p **and** = 2.5, p **or** = 1	.2608 (72.5)	.1706 (64.6)	.2600 (160.5)	.5359 (175.9)
p **and** = 2.5, p **or** = 2	.2557 (69.1)	.1663 (60.5)	.2514 (151.9)	.5104 (162.7)
p **and** = 3.5, p **or** = 1.2	.2587 (71.2)	.1682 (62.2)	.2524 (152.9)	.5295 (172.6)
p mixed (manual assignment)	.2592 (71.5)	.1705 (64.5)	-	-

Evaluation Summary for Mixed p-Value Assignment

(Precision Values for Recall of 0.25, 0.50, 0.75 averaged over
a Query Set for Each Collection)

Table 7

Recall	$p=\infty$ Boolean	$p=\infty$ weighted docs.	$p=1$ uniform	$p=2$ uniform	p **and** 2.5 p **or** 1.0
.1	.3430	.3856	.5153	.4979	.5077
.2	.2777	.3085	.4278	.4098	.4302
.3	.2064	.2498	.3543	.3500	.3586
.4	.2002	.2221	.3011	.2859	.2947
.5	.1757	.1999	.2453	.2443	.2455
.6	.1369	.1353	.1978	.1975	.1980
.7	.0648	.0733	.1488	.1491	.1478
.8	.0550	.0581	.1152	.1135	.1175
.9	.0401	.0416	.0874	.0860	.0873
1.0	.0371	.0360	.0559	.0567	.0571
average	.1537	.1710 (+ 11.0)	.2449 (+ 59.0)	.2391 (+ 56.0)	.2444 (+ 59.0)

a) CACM 3204, 52 Queries

Recall	$p=\infty$ Boolean	$p=\infty$ weighted docs.	$p=1$ uniform	$p=2$ uniform	p **and** 2.5 p **or** 1.0
.1	.2259	.2916	.3583	.3219	.3362
.2	.1951	.2397	.2865	.2811	.2865
.3	.1405	.1592	.2229	.2158	.2164
.4	.1202	.1312	.1781	.1777	.1825
.5	.0982	.1048	.1555	.1523	.1606
.6	.0800	.0808	.1330	.1323	.1400
.7	.0554	.0547	.1041	.1038	.1074
.8	.0482	.0478	.0842	.0845	.0836
.9	.0405	.0399	.0637	.0612	.0628
1.0	.0361	.0354	.0415	.0408	.0409
average	.1040	.1185 (+ 14.0)	.1628 (+ 57.0)	.1571 (+ 51.0)	.1617 (+ 55.0)

b) CISI 1460, 35 Queries

Recall-Precision Tables for Four Document Collections

Table 8

	$p=\infty$	$p=\infty$	$p=1$	$p=2$	p **and** 2.5
Recall	Boolean	weighted	uniform	uniform	p **or** 1.0
		docs.			
.1	.2509	.4322	.4960	.4746	.5072 ||
.2	.1723	.2971	.4215	.3988	.4310 ||
.3	.1416	.2233	.3503	.3421	.3534 ||
.4	.1227	.1850	.2924	.2866	.2968 ||
.5	.1057	.1373	.2480	.2445	.2514 ||
.6	.0719	.0881	.2069||	.2057	.2049
.7	.0439	.0471	.1611	.1661	.1677 ||
.8	.0362	.0359	.1285	.1288	.1320 ||
.9	.0095	.0098	.0734	.0757	.0773 ||
1.0	.0028	.0028	.0180	.0201	.0210 ||
average	.0958	.1459	.2396	.2343	.2443
		(+ 52.0)	(+ 15.0	(+ 145.0)	(+ 155.0)

c) Inspec 12684, 77 Queries

	$p=\infty$	$p=\infty$	$p=1$	$p=2$	p **and** 2.5
Recall	Boolean	weighted	uniform	uniform	p **or** 1.0
		docs.			
.1	.5528	.6339	.7956||	.7813	.7810
.2	.4313	.4609	.7174	.6966	.7256||
.3	.3065	.3398	.6885	.6713	.6988||
.4	.2370	.2582	.6219	.6034	.6288||
.5	.1630	.1866	.5286||	.5052	.5249
.6	.1532	.1779	.4729	.4599	.4749||
.7	.1065	.1259	.4064||	.3702	.3971
.8	.0769	.0926	.3578||	.3216	.3479
.9	.0381	.0525	.2219||	.2067	.2218
1.0	.0321	.0448	.1349||	.1043	.1278
average	.2097	.2373	.4946	.4721	.4929
		(+ 13.0)	(+ 136.0)	(+ 125.0)	(+ 135.0)

d) Medlars 1033, 30 Queries

Recall-Precision Tables for Four Document Collections

Table 8 (cont.)

Query ISI Q3	
Natural language:	"What is information science; give a definition if possible"
Original Boolean form:	(information **and** (science **or** definition))
Reformulated query:	((information **and**^{2} (science **or**$^{1.5}$ retrieval)) **or**$^{1.5}$ (automatic **and**$^{1.8}$ (documentation **or**$^{1.5}$ (library **and**$^{1.8}$ science))))
Phrases in reformulated query:	information science, information retrieval, automatic documentation, automatic library science
Query ISI Q2	
Natural language:	"How can pertinent data as opposed to references or entire articles be retrieved in answer to information requests"
Original Boolean form:	((data **or** information) **and** (automatic **or** retrieve **or** request **or** pertinent **or** response) **and not** (article **or** reference))
Reformulated query:	((automatic **and**$^{1.5}$ ((data **and**^{2} retreival) **or**$^{1.5}$ (question **and**^{2} answer) **or**$^{1.5}$ (passage **and**$^{1.5}$ retrieval) **or**$^{1.5}$ (data **and**$^{1.5}$ extraction))) **or**$^{1.5}$ ((data **and**^{2} base) **and**$^{1.8}$ (management **or**$^{1.5}$ retrieval)))
Phrases in reformulated query:	automatic data retrieval, automatic question answering, automatic passage retrieval, automatic data extraction, data base management, data base retrieval

Example of Manual Query Reformulation

Table 9

Original vs. Reformulated Queries	CACM 3204 52 queries		CISI 1460 35 queries	
	original	new forms	original	new forms
pure Boolean	.1512	.1688	.1037	.0845
$p = \infty$	.1720	.1873	.1147	.0921
(weighted docs.)	(+ 13.8)	(+ 10.9)	(+ 10.7)	(+ 9)
$p = 1$ uniform	.2594	.2819	.1681	.1802
	(+ 71.6)	(+ 67.0)	(+ 62.1)	(+ 113.3)
$p = 2$ uniform	.2558	.2731	.1676	.1800
	(+ 69.2)	(+ 61.8)	(+ 61.6)	(+ 113.1)
p **and** = 2	.2595	.2795	.1701	.1853
p **or** = 1.2	(+ 71.7)	(+ 65.6)	(+ 64.1)	(+ 119.4)
p **and** = 2.5	.2608	.2777	.1706	.1854
p **or** = 1	(+ 72.7)	(+ 64.6)	(+ 64.6)	(+ 119.4)

Summary of Averaged Output of Reformulated versus Original Queries for Two Collections

Table 10