

## THE NECESSITY FOR ADAPTATION IN MODIFIED BOOLEAN DOCUMENT RETRIEVAL SYSTEMS

MICHAEL D. GORDON\*

Computer and Information Systems, Graduate School of Business Administration,  
University of Michigan, Ann Arbor, MI 48109, U.S.A.

**Abstract**—A document retrieval system may be described by three formal characteristics: the syntax employed to describe documents (keywords or vectors of weights, for instance), the form of machine-processable queries it accepts as valid (unordered sets of keywords, keywords with Boolean connectives or weighted vectors, for example), and the retrieval rules used to rank or retrieve documents. This article argues that the interdependence among document descriptions, queries, and retrieval rules requires adaptation for the system to perform effectively when one of its components changes.

Recently, suggestions have been made to modify traditional Boolean document retrieval systems to allow more flexible queries and ranked document output. However, these new forms of queries and retrieval rules likely require that documents be described differently than they are in existing, commercial Boolean retrieval systems.

A "genetic algorithm" is discussed as a means for redescribing documents. This probabilistic algorithm uses feedback along with alternative descriptions of a single document and takes account of the dependency structure of subject terms.

### 1. INTRODUCTION

We find ourselves in somewhat of a dilemma: The commercial dominance of Boolean document retrieval systems has been accompanied by substantial investment, but these systems are deficient in performance in several ways. Practically, such investment demands these systems be used, even though they are not capable of supporting improved retrieval techniques reported in research laboratories. The challenge, then, is to improve the performance of these Boolean systems while keeping them operational.

In adjusting the way an information retrieval (IR) system operates, however, we may be tampering with the principles on which it is built. As we will see, a document description that suffices for traditional Boolean retrieval may be less than adequate if the operation of the system is altered. This article argues that adaptation should accompany the modification of operational Boolean systems and presents an adaptive technique that will effect such adjustment.

### 2. IR SYSTEMS VIEWED AS SYSTEMS

Document retrieval systems are comprised of three subsystems. These support querying by inquirers, the description of documents, and the selection or ranking of documents in response to a particular query (matching). We can look at each of these subsystems somewhat formally. Doing so, we see that the querying subsystem determines the form of machine-processable queries that will be accepted as valid (unordered sets of keywords, keywords with Boolean connectives, weighted term vectors, for example), the description subsystem determines the syntax employed to describe documents (keywords or vectors of weights, for instance), and the matching subsystem uses an algorithm either to select a subset of a document collection for the inquirer's inspection or to rank documents in terms of predicted relevance to a query.

A main point made in this article is that the three subsystems of an information retrieval system are interdependent. This implies that a change to the operation of one sub-

\*This research was supported by the Graduate School of Business Administration, Division of Research, University of Michigan.

system cannot be made independently of its effects on the other two. Instead, there is a certain context to an information retrieval system that encompasses its operation as a whole. Manual indexers, for instance, rarely assign just one or as many as 20 subject terms to documents. An experienced searcher, knowing this, expects that single-term Boolean queries will provide an overwhelming number of retrieved documents whereas lengthy ANDed queries will likely retrieve none. In this way, then, indexing, querying, and matching are somewhat in “balance” with each other. More generally, any document retrieval system will be most effective when there is proper balance among its three subsystems so that searchers have better success in retrieving the documents they desire. Such balance does not come automatically when a system is constructed. Instead, there is a great variety of information needs and perspectives that inquirers bring to a search, and there is no characterization of the subject content of documents with which all inquirers will agree. These two factors suggest that balance must be brought about by “tuning” the system by means of communication among its three components so that they work properly with each other. This phenomenon is similar to the evolution in natural language that allows communication by convention, rather than by fiat, as Cherry describes it [1].

A system whose operation is modified in any significant way (such as by changing the form of queries it accepts or changing the matching function it uses for selecting or ranking documents) is almost certainly out of balance. For such a system, communication among its subsystems is necessary so that adjustments in document descriptions, querying, or retrieval rules can be made, thus leading to improved, balanced retrieval.

We now articulate three principles concerning the interdependence of the subsystems of a document retrieval system. Each principle is illustrated by an example. Following the presentation of these principles and examples, we look at some suggested modifications to the operation of Boolean retrieval systems to see how these principles apply.

**EXAMPLE 1.** Consider the following collection of four documents indexed by five subject terms:

	$t_1$	$t_2$	$t_3$	$t_4$	$t_5$
doc <sub>1</sub>	1	1	0	0	1
doc <sub>2</sub>	0	1	0	0	1
doc <sub>3</sub>	1	1	0	0	0
doc <sub>4</sub>	0	1	0	0	0

Suppose that just the first two documents will be relevant to the information need of an inquirer. Then, this inquirer will achieve 100% recall by issuing the query

$$Q_1 = (t_1 \text{ OR } t_2) \text{ AND } t_5.$$

On the other hand, suppose the inquirer is uncertain about the extent to which  $t_1$  and  $t_2$  should be regarded as synonyms rather than co-occurring terms. If the inquirer issues the query

$$Q_2 = (t_1 \text{ AND } t_2) \text{ AND } t_5$$

instead of  $Q_1$ , recall will decline to 50%. Such uncertainty or confusion about representing an information need by a query is common [2]. Since only the query and not the descriptions has changed, we are led to the following:

*Principle 1.* With respect to retrieval performance, the adequacy of a document description depends on the queries used to retrieve the document that the description represents. Or, change users' queries, and you may need to modify document descriptions in compensation.

EXAMPLE 2. Let the following three binary vectors represent a query and two documents, respectively.

	$t_1$	$t_2$	$t_3$	$t_4$	$t_5$	$t_6$	$t_7$	$t_8$
query	1	1	1	0	0	0	0	0
doc <sub>1</sub>	0	1	1	1	1	0	0	0
doc <sub>2</sub>	1	1	1	1	1	1	1	1

Suppose that the first document would be judged more relevant to the user's need (as expressed by his or her query) than would the second. With a similarity-based (rather than Boolean) matching function, the first document should receive a better ranking (higher similarity to the query) than the second. Using Jaccard's matching score, this relationship does, indeed, apply. That is,

$$\text{Jac}(\text{query}, \text{doc}_1) = 0.40 > \text{Jac}(\text{query}, \text{doc}_2) = 0.375.$$

( $\text{Jac}(X, Y)$  is the ratio of the number of elements in common to sets  $X$  and  $Y$  to the total number of elements in the union of the two sets. Here  $X$  is a set of query terms,  $Y$  a set of document terms.)

On the other hand, if we use the cosine score as our basis of similarity, we see that the relative order of the two rankings is reversed. That is,

$$\text{Cos}(\text{query}, \text{doc}_1) = 0.577 < \text{Cos}(\text{query}, \text{doc}_2) = 0.612.$$

This leads us to:

*Principle 2.* For a given query, the adequacy of a document description depends on the matching function used to retrieve (or rank) documents. Or, change the matching function, and you may need to modify document descriptions as well.

EXAMPLE 3. Consider two ways of indexing the same document: (1) automatically, by using all the nontrivial terms in its abstract or (2) intellectually, by indexers selecting terms from a controlled vocabulary. Katzer et al. [3] performed experiments which compared seven different methods of indexing in common use, including the two just mentioned. The effectiveness of each of the methods was fairly similar, but each form of document representation tended to retrieve different relevant documents for the same query. A study by Tenopir reported similar findings [4]. These observations confirm:

*Principle 3.* The form of description of a document influences retrieval performance for a given query and matching function.

We now examine some proposals concerning ways in which Boolean retrieval systems might be modified and see how these principles apply. The proposals that follow do not constitute a complete set of suggested modifications to Boolean retrieval systems.

### 3. MODIFICATIONS OF BOOLEAN RETRIEVAL

The conceptual basis for Boolean retrieval predates the computer. But, by efficiently performing set-theoretic based operations on inverted lists of index terms, the computer has become the ideal mechanism to implement Boolean retrieval. Altogether, information retrieval systems based on Boolean retrieval (particularly those using the full text of a document as a basis for representation) perform creditably, if unspectacularly [4]. Because of the terrific head start Boolean retrieval systems have enjoyed over systems based on other principles, together with their creditable performance, these systems dominate the market commercially. As a result, significant investment has been made in terms of hardware, software, and data entry to support both the large subscription information retrieval databases

and automated library systems on which many information workers and people in various businesses and professions already rely. Having captured such a large share of the commercial document retrieval market, Boolean systems face little near term threat from competing document retrieval models.

Nonetheless, Boolean retrieval systems are limited. Most notably, the binary logic these systems employ can make it difficult to identify precisely a set of relevant documents. Inquirers too often find they end up with enormous sets of retrieved documents (and very bad precision) or more manageable retrieved sets and very poor recall. Since the binary logic that underlies Boolean retrieval means that a document is either selected for retrieval or is not, output presented to the inquirer in order of decreasing likelihood of relevance is impossible, except for saying that documents selected are considered by the system to be relevant whereas those that are not selected are considered to be nonrelevant. Furthermore, a document exhibiting "partial similarity" to a query will not be brought to the inquirer's attention, such as a document indexed by  $t_1$ ,  $t_2$ ,  $t_3$ , and  $t_4$  in response to the query  $t_1$  AND  $t_2$  AND  $t_3$  AND  $t_4$  AND  $t_5$ . Other difficulties with Boolean systems include the inquirer's inability to weight the importance of query terms, the difficulty some people have in understanding the proper syntactic use of Boolean AND and Boolean OR, and the difficulty in sometimes determining whether two Boolean expressions should be regarded as synonyms (and thus ORed) rather than conjuncts (and thus ANDed).

In response to these difficulties, the basic operation of Boolean systems has been extended. One form of extension follows a straightforward Boolean match to a query by ranking the documents thus retrieved [5,6]. In this article, we consider suggestions of this type as well as other suggestions for more fundamentally modifying Boolean retrieval systems. Our examination of these suggestions will focus on the principles we have described above.

Because of the interdependence of the three subsystems in a document retrieval system, some of the changes we will examine involve two subsystems (usually querying and matching). Nonetheless, in our analysis we try to categorize changes in terms of the subsystem that is altered most significantly.

### 3.1 *Changes to querying*

As we have suggested, Boolean queries present certain difficulties. As a consequence, certain suggestions have been made to alter this aspect of retrieval. One type of modification has been suggested by Bookstein for use in fuzzy retrieval [7]. Since Boolean retrieval is a special case of fuzzy retrieval, this suggestion has potential application to Boolean retrieval. This type of modification permits the use of weights to be attached to Boolean request terms. Thus, instead of being limited to requests similar to  $(t_1$  OR  $t_3)$  AND  $t_4$ , one can emphasize the relative importance of these terms by requests of this form:  $(t_1[.5]$  OR  $t_3[.7])$  AND  $t_4[1.0]$ . As a result, one can retrieve documents that are ranked according to predicted decreasing relevance.

A somewhat similar approach to modifying the operation of Boolean retrieval systems has been suggested by Paice, who discusses the uncertainty concerning the selection of Boolean operators, particularly in complex Boolean expressions [8]. As a result, Paice relaxes the traditional use of Boolean operators, allowing AND to behave a bit more like OR and vice versa. That is, conjunctive (ANDed) queries may retrieve a document whose representation fails to use all ANDed terms, and queries involving ORed terms can favor those documents whose representations use more of those terms. Under this model, one is permitted to make requests resembling:  $(t_1$  OR[.4]  $t_3)$  AND[.6]  $t_4$ . As in the previous approach, the output of documents is rank ordered.

Even though these modifications may be judged by considering their effectiveness in laboratory or real-life retrieval settings, we also see that our first principle applies. Recall that this principle states that the adequacy of a document description depends on the queries used to retrieve it. The descriptions in a document collection used for strict Boolean retrieval can likely be altered to perform more effectively with a retrieval model employing some other form of querying, such as those that Bookstein and Paice propose. Even

if a modified Boolean system performs effectively when tested, adaptation may improve performance further. Suppose, for example, we could derive some theoretical (and feasible) principles or useful heuristics for weighting query terms or Boolean operators in a modified Boolean retrieval system. These rules might resemble: Weight query terms by importance, the most important being given weight 1.0, the next most important weight 0.8, etc. Or, the best mix for weighting ANDs and ORs is AND(.67), OR(.33). However, we still might ask: can retrieval be even better? Were the documents being described with these rules in mind? Clearly, they were not. It is probable, then, that we can better tailor document descriptions to be used with these rules.

Our goal in document retrieval is not so much to find the best way to articulate queries (or describe documents or perform a matching function) but, rather, to find the best ways to retrieve documents. Document descriptions interact with the other two retrieval subsystems (querying and matching) much the same way the dosage of a drug interacts with a patient's weight to influence the patient's response. Should the patient's weight change, a new dosage of the drug may be needed. Similarly, change the way queries are made, and revised document descriptions may be advised to bring the retrieval system into better balance.

### 3.2 Changes to matching

Recent thought has focused on generalized models of Boolean retrieval that take advantage of the AND, OR, NOT query structure but try to overcome the shortcomings of traditional Boolean retrieval. This has been done from the vantage of probability theory [9] and also by methods which we view as changing the Boolean matching function. We briefly explore the latter approach here.

Under a strict Boolean interpretation, the request  $t_1$  AND ( $t_2$  OR  $t_3$ ) only retrieves documents employing  $t_1$ , subject to the condition that they employ at least one of the terms  $t_2$  or  $t_3$  in their description as well. A complete relaxation of the standard Boolean interpretation of AND and OR, on the other hand, can mean that, due to an identical number of common document-query terms, a document using just terms  $t_2$  and  $t_3$  in its description is predicted to be equally likely to be relevant as one employing just terms  $t_1$  and  $t_2$ . Systems employing  $p$ -norm retrieval, described by Salton, Fox, and Wu, allow Boolean matching to behave anywhere between these two extremes [10]. Boolean operators can be made to obey complete standard Boolean logic (when parameter  $p$  has a value of infinity), to disregard completely these Boolean connectives (by giving  $p$  the value 1) or to blur the distinction between AND and OR to varying degrees (by selecting values of  $p$  between 1 and infinity). This method has been shown to be an improvement over standard Boolean retrieval in laboratory settings.

However, the use of  $p$ -norm governed matching without redescribing documents calls to mind Principle 2: For a given query, the adequacy of a document description depends on the matching function used to retrieve (or rank) documents. Recall that the example we used to illustrate this principle showed documents ranked differently depending on which matching rule was employed, an occurrence that is common in retrieval experiments.

Again, if we make an adjustment to the matching function in order to change the usual operation of Boolean connectives, we should also seek to improve the document descriptions with which this new retrieval model will work. (Experience with  $p$ -norm retrieval shows that different values of the parameter  $p$  are most effective with different document collections. However, greater sensitivity in improving retrieval should be possible by adjusting document descriptions than by adjusting a small number of retrieval parameters.) What this article has been trying to show is that it is mistaken to say simply that a document is "about some subject." Rather, the description of a document should be cast in operational terms: with respect to getting this document into the hands of those who will find it useful, and keeping it from those who will not, how should this document be described? This is the principle embodied by Maron and Kuhns' notion of probabilistic indexing [11] and the more general decision theoretic notion of indexing described by Cooper and Maron [12] and others.

## 4. ADAPTATION OF DESCRIPTIONS

We have seen that changes to Boolean retrieval systems that result in new methods of querying or new methods of matching should be accompanied by adjustments in document descriptions. In this section, we make some final comments about adaptation and then describe a “genetic” algorithm that can be used with Boolean-based retrieval systems to improve document descriptions.

The idea of adaptation within the context of an information retrieval system is not new. Salton’s book on the SMART retrieval system, for instance, includes research contributions concerning ways to modify both document descriptions and queries in a cosine similarity-based retrieval environment [13]. The basic idea underlying these, and any, techniques of adaptation is this: If an acceptably working system is suspected of being able to perform better, use feedback to make some small changes to improve its performance. Thus, gradual changes are made to queries or document descriptions to bring about these improvements. This is analogous to adding a little spice to improve an already acceptable recipe, or “trueing” the spokes on a bicycle wheel that is a little out of adjustment simply to make it roll more smoothly.

We can call such adjustments “adaptation within the system,” to suggest that we are improving the performance of something that is essentially set up properly. Changes in language or the addition of new documents to a collection may require adaptation within a document retrieval system. Such adaptation can be effected by adjusting document descriptions periodically. Adaptation within a Boolean system is possible in other ways, too. Radecki’s proposal to take the output generated by standard Boolean methods and then probabilistically rank retrieved documents by classes can be regarded as adaptation within the system, since it is predicated on the notion that Boolean retrieval generally works but needs some enhancement [6]. Similarly, it may be accurate to regard certain Boolean front ends as adaptations within the system, for they operate on the assumption that Boolean methods are generally acceptable but must provide some user assistance to be truly effective. Blair, for instance, proposes a method for helping inquirers overcome a psychological “fixedness” in querying [14]. Heine proposes assisting the Boolean searcher by predicting the likely effectiveness of various Boolean queries involving terms the user supplies [15].

“Adaptation of the system” suggests a stronger need for adjustment than does “adaptation within the system.” It is the adaptation necessary to establish proper intercoordination of the three retrieval subsystems after one of them has been altered in some fundamental way (for instance by changing a system’s basic retrieval rules). It is also warranted in adjusting a system which has never been changed but has been designed without proper balance. We conclude this article with a brief description of a particularly powerful algorithm for document redescription – the genetic algorithm – which may be especially useful for providing adaptation of a modified Boolean system.

The genetic algorithm can be used with any retrieval model requiring adaptation of document descriptions, Boolean or not. This form of adaptation requires that a set of complete descriptions be associated with any one document. Retrieval of a document with multiple descriptions occurs by matching separately each of its descriptions with a query and arriving at some sort of “compromise” retrieval decision (perhaps the average similarity score of the separate matches in similarity-based retrieval; or a retrieve/no retrieve decision determined in majority rules fashion in a strict Boolean system). The crux of this probabilistic algorithm is to set up a competition among descriptions and to allow the ones performing best to more strongly influence retrieval in the future. Details of the algorithm are found in Holland [16]. Its application to information retrieval and its success in promoting (non-Boolean) retrieval are described in Gordon [17,18]. A comparison with probabilistic models is described in Gordon [19], in which genetic adaptation is shown to take account of the statistical dependency among index terms. We focus here on describing a data structure that will support this algorithm in retrieval systems using inverted lists to describe documents. Thus, this data structure is suitable for adaptation of either standard or modified Boolean retrieval systems implemented by inverted files.

The data structure needed to support genetic adaptation with (modified or standard) Boolean retrieval is easily illustrated by example. Suppose document<sub>x</sub> is represented logically by the following set of four “competing” descriptions. (Recall that the algorithm requires that a set of descriptions be associated with a document.)

	<i>t</i> <sub>1</sub>	<i>t</i> <sub>2</sub>	<i>t</i> <sub>3</sub>	<i>t</i> <sub>4</sub>	<i>t</i> <sub>5</sub>
desc <sub>x1</sub> =	⟨1,	1,	0,	0,	0⟩
desc <sub>x2</sub> =	⟨1,	0,	1,	0,	0⟩
desc <sub>x3</sub> =	⟨1,	1,	1,	0,	0⟩
desc <sub>x4</sub> =	⟨0,	1,	1,	0,	1⟩

Internally, the inverted list representations of the same descriptions would be similar to  $t_1 = \{\text{desc}_{x1}, \text{desc}_{x2}, \text{desc}_{x3}\}$ ,  $t_2 = \{\text{desc}_{x1}, \text{desc}_{x3}, \text{desc}_{x4}\}$ ,  $t_3 = \{\text{desc}_{x2}, \text{desc}_{x3}, \text{desc}_{x4}\}$ ,  $t_4 = \{\}$ ,  $t_5 = \{\text{desc}_{x4}\}$ . (Of course, each of these sets would also contain other document descriptions using the given index term.) Each element, desc<sub>xi</sub>, of any of these sets is actually a pointer to the text of (or, more probably, reference to) document<sub>x</sub>—just as it would be in a standard inverted file implementation. However, for genetic adaptation to take place, the text of (or reference to) document<sub>x</sub> must point (lead) to all complete descriptions of the document along with certain statistics for each of these descriptions—statistics that are updated as in the following example (see Fig. 1). Note the changes to traditional inverted file implementations this requires: (1) the addition of a pointer from a document’s text (or a reference to it) to the set of complete (noninverted) descriptions of that document; and (2) the inclusion of these noninverted descriptions in the database along with statistics concerning their retrieval effectiveness. Neither of these changes interferes in any significant way with a standard inverted list representation.

**EXAMPLE 4.** Suppose an inquirer issues the query  $t_1$  AND  $t_3$ . Following the intersection of sets  $t_1$  and  $t_3$  (which are, respectively, a set of pointers to documents indexed by  $t_1$ , and a set of pointers to documents indexed by  $t_3$ ), the documents referred to by the pointers in this intersection will be retrieved. The pointers desc<sub>x2</sub> and desc<sub>x3</sub> will both be in this intersection. Since these pointers both refer to the same document, that document (document<sub>x</sub>) will be retrieved just once for inspection. The statistics associated with each of the separate descriptions of the document will be updated according to the inquirer’s relevance assessment. Supposing the inquirer finds this document relevant, then desc<sub>x2</sub> and desc<sub>x3</sub> will both get credit for a “hit” (true positive). The other two descriptions shown, failing to match the query when they should have, will get credit for a false negative. (When a document is retrieved but found not relevant, each of its descriptions is credited either with a false positive or a true negative if the description matches or does not match the query, respectively. See Fig. 1 again.) After a specified number of retrievals of document<sub>x</sub>, the statistics associated with the four competing descriptions of the document are used by the genetic algorithm to replace the prevailing set of descriptions by another set of four descriptions better adapted to the queries already considered by the system. (In actual implementation, a document would probably require 10 to 15 descriptions rather than 4.) As inquirers begin to change the language by which they search for a document, its description changes as well.

Three final comments about the implementation we have described are in order. One, each alternative description of a document will generally contain far fewer terms than the document’s full-text representation. Two, the alternative descriptions of a document can be generated in a number of ways. For example, an existing description of a document (even its full-text description), together with certain probabilistic assumptions concerning the terms it uses (plus related terms), can be used to generate automatically alternative descriptions. Three, the secondary storage required for additional document descriptions will increase but not prohibitively. By describing a document by four descriptions, we incur a linear (fourfold) increase in the number of pointers needed, assuming all descriptions contain approximately the same number of terms. Each noninverted description of a doc-

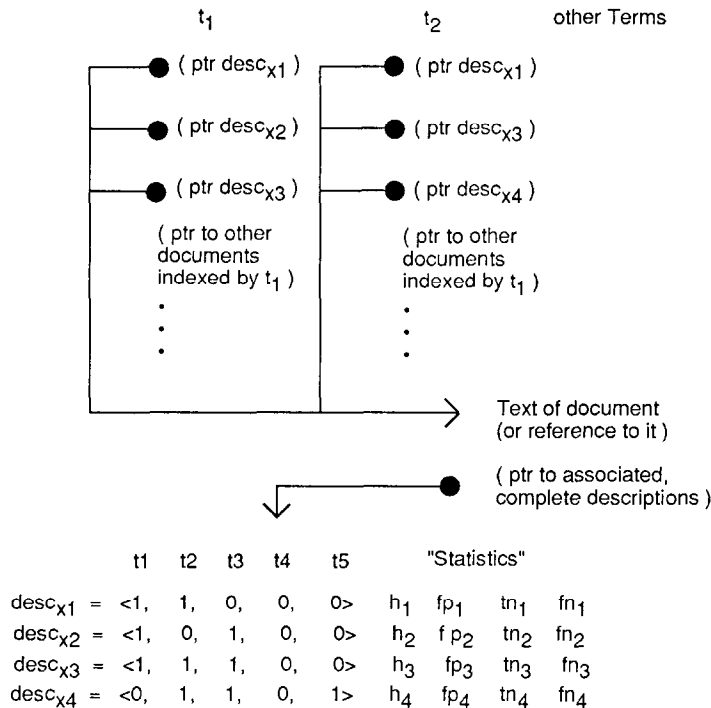


Fig. 1. Data structure for genetic adaptation with inverted lists. This figure shows how several descriptions of the same document can be used in an inverted file system supporting genetic adaptation. Note, for instance, that three descriptions of document<sub>x</sub> employ  $t_1$ . Observe also that the text of (or reference to) document<sub>x</sub> leads to the set of complete document descriptions and statistics. Document statistics reveal hits ( $h$ ), false positives ( $fp$ ), true negatives ( $tn$ ), and false negatives ( $fn$ ) for each description. These statistics are used by the genetic algorithm to produce better document descriptions.

ument can be compactly represented, and document description statistics require little storage. If the full text of the document is actually stored (in contiguous, readable fashion) the increase in storage incurred by needing extra pointers (because of multiple document descriptions) is negligible. An increased cost in storage is likely outweighed by the improvements in retrieval performance genetic adaptation can provide, and dropping costs of secondary storage further argue that increased storage can be tolerated.

## 5. CONCLUSION

An argument has been made that querying, matching, and document description cannot be regarded as unrelated activities but, rather, must be interrelated. When standard Boolean systems are modified, this argument implies that some sort of adaptation be performed (to document descriptions) to restore a proper interrelationship. An adaptive algorithm for redescribing documents has been mentioned as a candidate governing such adaptation, and its physical implementation by an inverted file system has been described.

## REFERENCES

1. Cherry, C. On human communication: A review, a survey, and a criticism. Third edition. Cambridge, MA: MIT Press; 1978.
2. Borgman, C.L. Why are online catalogs hard to use? Lessons learned from information-retrieval studies. *Journal of the American Society for Information Science*, 37(6): 387-400; 1986.
3. Katzer, J.; McGill, M.J.; Tessier, J.A.; Frakes, W.; DasGupta, P. A study of the overlap among document representations. *Information Technology: Research and Development*, 1(4): 261-274; 1982.
4. Tenopir, C. Full text database retrieval performance. *Online Review*, 9(2): 149-164; 1985.
5. Noreault, T.; Koll, M.; McGill, M.J. Automatic ranked output from Boolean searches in SIRE. *Journal of the American Society for Information Science*, 28(6): 333-339; 1977.



6. Radecki, T. Incorporation of relevance feedback into Boolean retrieval systems. In: Salton, G.; Schneider, H.-J., editors. *Research and development in information retrieval*. Berlin: Springer-Verlag; 1983: 133-150.
7. Bookstein, A. Fuzzy requests: An approach to weighted Boolean searches. *Journal of the American Society for Information Science*, 31(4): 240-247; 1980.
8. Paice, C.D. Soft evaluation of Boolean search queries in information retrieval systems. *Information Technology: Research Development Applications*, 3(1): 33-41; 1984.
9. Bookstein, A. Implications of Boolean structure for probabilistic retrieval. *Proceedings of the eighth annual international ACM SIGIR conference on research and development in information retrieval*; 1985 June 5-7; Montreal.
10. Salton, G.; Fox, E.A.; Wu, H. Extended Boolean information retrieval. *Communications of the ACM*, 26(11): 1022-1036; 1983.
11. Maron, M.E.; Kuhns, J.L. On relevance, probabilistic indexing and information retrieval. *Journal of the ACM*, 7(3): 216-244; 1960.
12. Cooper, W.S.; Maron, M.E. Foundations of probabilistic and utility-theoretic indexing. *Journal of the ACM*, 25(1): 67-80; 1978.
13. Salton, G., editor. *The SMART retrieval system—Experiments in automatic document processing*. Englewood Cliffs, NJ: Prentice-Hall; 1971.
14. Blair, D.C. Searching biases in large interactive document retrieval systems. *Journal of the American Society for Information Science*, 31(4): 271-277; 1980.
15. Heine, M.H. A simple, intelligent front end for information retrieval systems using Boolean logic. *Information Technology: Research and Development*, 1(4): 247-260; 1982.
16. Holland, J. *Adaptation in natural and artificial systems*. Ann Arbor, MI: University of Michigan Press; 1975.
17. Gordon, M. *Adaptive subject description in document retrieval*. Ph.D. thesis, University of Michigan; Ann Arbor, MI; 1984.
18. Gordon, M. A learning algorithm applied to document redescription. *Proceedings of the eighth annual international ACM SIGIR conference on research and development in information retrieval*; 1985 June 5-7; Montreal.
19. Gordon, M. *Probabilistic and genetic algorithms in document retrieval*. Working Paper No. 450, Division of Research, Graduate School of Business Administration, University of Michigan; November 1985.