INFORMATION TO USERS

This was produced from a copy of a document sent to us for microfilming. While the most advanced technological means to photograph and reproduce this document have been used, the quality is heavily dependent upon the quality of the material submitted.

The following explanation of techniques is provided to help you understand markings or notations which may appear on this reproduction.

1. The sign or "target" for pages apparently lacking from the document photographed is "Missing Page(s)". If it was possible to obtain the missing page(s) or section, they are spliced into the film along with adjacent pages. This may have necessitated cutting through an image and duplicating adjacent pages to assure you of complete continuity.

2. When an image on the film is obliterated with a round black mark it is an indication that the film inspector noticed either blurred copy because of movement during exposure, or duplicate copy. Unless we meant to delete copyrighted materials that should not have been filmed, you will find a good image of the page in the adjacent frame. If copyrighted materials were deleted you will find a target note listing the pages in the adjacent frame.

3. When a map, drawing or chart, etc., is part of the material being photographed the photographer has followed a definite method in "sectioning" the material. It is customary to begin filming at the upper left hand corner of a large sheet and to continue from left to right in equal sections with small overlaps. If necessary, sectioning is continued again—beginning below the first row and continuing on until complete.

4. For any illustrations that cannot be reproduced satisfactorily by xerography, photographic prints can be purchased at additional cost and tipped into your xerographic copy. Requests can be made to our Dissertations Customer Services Department.

5. Some pages in any document may have indistinct print. In all cases we have filmed the best available copy.

University Microfilms International
300 N. ZEEB RD., ANN ARBOR, MI 48106
Brinkman, Barry John

THE DEVELOPMENT AND EVALUATION OF INTERACTIVE AIDS FOR SEARCH PROFILE CONSTRUCTION IN DOCUMENT RETRIEVAL SYSTEMS

The Ohio State University

Ph.D. 1982

University
Microfilms
International
300 N. Zeeb Road, Ann Arbor, MI 48106
THE DEVELOPMENT AND EVALUATION
OF INTERACTIVE AIDS FOR SEARCH PROFILE CONSTRUCTION
IN DOCUMENT RETRIEVAL SYSTEMS

DISSERATION

Presented in Partial Fulfillment of the Requirements for
the Degree of Doctor of Philosophy in the Graduate
School of The Ohio State University

By
Barry John Brinkman, B.S., M.S.

The Ohio State University
1982

Reading Committee:  Approved By
Anthony E. Petrarca  
Lee J. White  
Ronald L. Ernst  

Adviser
Department of Computer and
Information Science
I wish to express my sincere thanks and appreciation to each of the many people who have contributed to the completion of this dissertation. In particular, my adviser, Dr. Anthony Petrarca, has provided excellent guidance throughout the course of this research. The other members of my reading committee, Dr. Lee White and Dr. Ronald Ernst, have made many helpful suggestions.

Dr. Jerry Lazorick introduced me to the field of information storage and retrieval. Dr. Tom Bishop of Battelle Columbus Labs examined the statistical design techniques which I was considering and made several very helpful suggestions in this area of the research.

Without the contribution and support of Bern Bayer, Manager of the Mechanized Information Center of the Ohio State University Libraries, and his entire staff this study could not have been performed. Ross Poli, Marty Goldsmith, Poly Bean, Noelle vanPulis, and Larry Perk gave generously of their time which is extremely limited by the demands of their jobs. They all helped gather the necessary data and made valuable suggestions while I was designing the Standard system. The present and past programming staff, Russ Hand, John Hsu, Ron Beaton, and Conchita Beaton patiently
provided the necessary data files, often on short notice, and explained many of the finer points of the MIC software and file structures.

Ken Szczesny, Bill Griffith, and John Fried of Battelle Columbus Labs have been most patient and understanding in allowing me the flexibility in my working hours necessary to gather the data and complete the writing of this dissertation.

A special thanks is extended to Cheri Erkenbrecher who typed several drafts of this document at very inconvenient times, and who also typed most of the final copy. Sue Rhea helped to finish typing the final copy.

The support and encouragement provided by my parents, Mr. and Mrs. Harry Brinkman, and by my wife's parents, Mr. and Mrs. Tim Murphy, throughout the course of this research is greatly appreciated.

Finally, I extend very sincere thanks and appreciation for the patience, understanding, and support of my wife, Pat. Without her financial, emotional, and sometimes physical support of the entire effort, this dissertation would not have been possible.
VITA

December 13, 1949 . . . Born - Worthington, Ohio

1971 . . . . . . . . B.S., University of Notre Dame, Notre Dame, Indiana

1974 . . . . . . . . M.S., The Ohio State University, Columbus, Ohio

1971 - 1979 . . . . Research and Teaching Associate, The Ohio State University, Columbus, Ohio

1979 - 1982 . . . . Research Scientist, Battelle Columbus Laboratories, Columbus, Ohio
FIELDS OF STUDY

Major Field: Computer and Information Science

Studies in Information Storage and Retrieval and Information Systems.
Dr. Anthony E. Petrarca and Dr. Gerald J. Lazorick

Studies in Computer Programming and Systems Programming.
Dr. Clinton R. Foulk

Studies in Operations Research
Dr. Wilbert E. Wilhelm
TABLE OF CONTENTS

Page

ACKNOWLEDGEMENTS ............................................... ii

VITA .......................................................... iv

LIST OF TABLES ................................................. ix

LIST OF FIGURES ............................................... xi

CHAPTER

I. INTRODUCTION ........................................... 1

II. DOCUMENT RETRIEVAL SYSTEMS - AN OVERVIEW ............. 5

2.1 A Brief History ........................................ 5
2.2 Who Should Interact with the System .................. 8
2.3 Online Search Aids - Background ..................... 11
2.4 Current Online Search Aids ............................ 14
2.5 Association Measures and Their Computation .......... 15
2.6 Experimental Aids Based on Term-Term and
    Document-Document'Associations ..................... 25
2.7 Profile Retrieval Aids ................................ 28
2.8 The SMART System's Relevance Feedback ............. 30

III. THE EVALUATION OF RETRIEVAL SYSTEMS AND
     THEIR COMPONENTS .................................. 32

3.1 Precision, Recall and Other Measures ................ 32
3.2 Problems Associated with Retrieval
    System Comparison .................................. 36
3.3 A Single Scale for Comparing Retrieval
    System Output ...................................... 37
3.4 Proposed Model for Comparative Evaluation
    of Retrieval Outputs ............................... 39
IV. THE ENVIRONMENT FOR THE EXPERIMENT ........................................... 49

4.1 The Mechanized Information Center and Its Retrieval Services ................. 49

4.1.1 MIC Databases .................................................. 50

4.1.2 MIC System Software and File Structures ..................................... 51

4.1.3 MIC Query Language and Profile Structure ..................................... 54

4.2 The Standard Online System ...................................................... 59

4.3 The Standard System Versus the State of the Art ................................ 64

V. THE AIDS SYSTEM AND THE INDIVIDUAL AIDS ........................................ 66

5.1 The Concepts Behind the Aids ...................................................... 66

5.2 The Profile-Profile Association Aid ................................................. 69

5.3 The Term-Term (Profile) Association Aid ........................................... 76

5.4 The Term-Term (Document) Association Aid ......................................... 80

5.5 The Relevance Feedback Aid ............................................................ 82

5.6 The Aids System - User's View ....................................................... 85

VI. EXPERIMENTAL DESIGN OF EVALUATION PROCEDURES ................................ 89

6.1 Design Considerations ........................................................................ 89

6.2 Design Alternatives ............................................................................ 92

6.2.1 Alternative 1 - Total User Involvement ............................................ 92

6.2.2 Alternative 2 - Limited User Involvement ....................................... 98

6.3 Design Alternative Chosen for the Experiment ..................................... 100

6.4 Evaluation of the Individual Aids ....................................................... 102

VII. EXPERIMENTAL RESULTS ..................................................................... 105

7.1 The Evaluation of the Aids System ..................................................... 105

7.1.1 Gathering of the Evaluation Data .................................................... 105

7.1.2 Statistical Tests for Significance .................................................... 108

7.1.3 Analysis and Treatment of Ties ....................................................... 111

7.1.4 Profile Preparation Time Considerations ......................................... 112

7.1.5 Computer Overhead Considerations ................................................. 115

7.1.6 Impact of Omitted State of the Art Features on the Standard and Aids Systems ...................................................... 116

7.1.7 Comparison of Observed System Preferences With Those Predicted by the Utility Model ...................................................... 117

7.2 Evaluation of the Individual Aids ....................................................... 118

VIII. CONCLUSIONS AND DIRECTIONS FOR FUTURE RESEARCH ..................... 122
APPENDICES

A. MIC-ONLINE User's Manual .................................. 128
B. File Structures Used for the Aids Program and Algorithms Used to Invoke the Aids ................. 138
C. System Description of the Construction of the Aids .................................................. 145
D. Use of MIC Weighting Formula in a Predictable Manner ............................................... 157

BIBLIOGRAPHY ........................................................................ 161
GLOSSARY ................................................................................. 166
**LIST OF TABLES**

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Contingency Table of Retrieval Results</td>
<td>33</td>
</tr>
<tr>
<td>3.2</td>
<td>Retrieval Evaluation Measures</td>
<td>34</td>
</tr>
<tr>
<td>4.1</td>
<td>A Few Examples of Coding a Profile in MIC Format</td>
<td>58</td>
</tr>
<tr>
<td>5.1</td>
<td>Example of Term-Term (Profile) Association Values</td>
<td>77</td>
</tr>
<tr>
<td>5.2</td>
<td>An Example of Term-Term (Document) Association Values</td>
<td>83</td>
</tr>
<tr>
<td>7.1</td>
<td>Evaluation Data for Comparison of the Standard System</td>
<td>109</td>
</tr>
<tr>
<td></td>
<td>with the Aids System</td>
<td></td>
</tr>
<tr>
<td>7.2</td>
<td>A Summary of the Evaluation Data From Table 7.1</td>
<td>110</td>
</tr>
<tr>
<td>7.3</td>
<td>Summary of the Statistical Test for Significance</td>
<td>110</td>
</tr>
<tr>
<td></td>
<td>Excluding Ties</td>
<td></td>
</tr>
<tr>
<td>7.4</td>
<td>Summary of the Statistical Test for Significance</td>
<td>113</td>
</tr>
<tr>
<td></td>
<td>Including Ties</td>
<td></td>
</tr>
<tr>
<td>7.5</td>
<td>Performance of the Term-Term Association Aids</td>
<td>119</td>
</tr>
<tr>
<td></td>
<td>on a Term Efficiency Basis</td>
<td></td>
</tr>
<tr>
<td>7.6</td>
<td>Performance of the Individual Aids on a Profile</td>
<td>120</td>
</tr>
<tr>
<td></td>
<td>Efficiency Basis</td>
<td></td>
</tr>
<tr>
<td>B.1</td>
<td>Files used by the Aids Program</td>
<td>143</td>
</tr>
<tr>
<td>C.1</td>
<td>Description of the Files used to Construct the Aids</td>
<td>149</td>
</tr>
<tr>
<td></td>
<td>Based on the Profiles</td>
<td></td>
</tr>
<tr>
<td>C.2</td>
<td>Description of the Files used to Construct the Aid</td>
<td>154</td>
</tr>
<tr>
<td></td>
<td>Based on the Documents</td>
<td></td>
</tr>
<tr>
<td>Table</td>
<td>Page</td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td>D.1</td>
<td>Table of Function Values for Some Specific Values of $w_1$ and $w_2$ for a Two Term Match</td>
<td>158</td>
</tr>
<tr>
<td>D.2</td>
<td>Table of Function Values for Some Specific Values of $w_1$, $w_2$ and $w_3$ for a Three Term Match</td>
<td>159</td>
</tr>
<tr>
<td>D.3</td>
<td>Two Examples of Coding a Profile using the Generalized Approximation Formula</td>
<td>160</td>
</tr>
</tbody>
</table>
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Association Measure Computation Examples</td>
<td>18</td>
</tr>
<tr>
<td>2.2</td>
<td>Associative Retrieval - An Example</td>
<td>20</td>
</tr>
<tr>
<td>2.3</td>
<td>Term-Term Association Matrix (B) Computed from Document-Term Matrix (C) in Figure 2.2 Using the Tanimoto Association Measure</td>
<td>22</td>
</tr>
<tr>
<td>2.4</td>
<td>Document-Document Association Matrix (A) Computed from Document-Term Matrix (C) in Figure 2.2 Using the Tanimoto Association Measure</td>
<td>22</td>
</tr>
<tr>
<td>2.5</td>
<td>Document-Term Matrix (General Case)</td>
<td>23</td>
</tr>
<tr>
<td>2.6</td>
<td>A Document-Term Network</td>
<td>28</td>
</tr>
<tr>
<td>3.1</td>
<td>The Problem of Dividing the R/P Space</td>
<td>41</td>
</tr>
<tr>
<td>3.2</td>
<td>An Example of Dividing the R/P Space</td>
<td>41</td>
</tr>
<tr>
<td>3.3</td>
<td>Utility Graph for Comparative Evaluation of Retrieval Outputs (Full Retrospective File Scale)</td>
<td>44</td>
</tr>
<tr>
<td>3.4</td>
<td>Utility Graph for Comparative Evaluation of Retrieval Outputs (Test File Scale)</td>
<td>47</td>
</tr>
<tr>
<td>4.1</td>
<td>Structure of the Original MIC Multidisciplinary Current Awareness System</td>
<td>53</td>
</tr>
<tr>
<td>4.2</td>
<td>A Sample MIC Notification Card</td>
<td>55</td>
</tr>
<tr>
<td>4.3</td>
<td>A Sample Terminal Session - Standard System</td>
<td>63</td>
</tr>
<tr>
<td>5.1</td>
<td>Computing Profile-Profile Associations Using the Tanimoto Measure (Equation 5.1)</td>
<td>71</td>
</tr>
<tr>
<td>5.2</td>
<td>Computing Profile-Profile Associations Using the Asymmetric Subset Measure (Equation 5.2) for the Profiles Defined in Figure 5.1</td>
<td>73</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td>------</td>
</tr>
<tr>
<td>5.3</td>
<td>An Example of the Relevance Feedback Aid</td>
<td>84</td>
</tr>
<tr>
<td>5.4</td>
<td>A Sample of the Term-Term Aids</td>
<td>86</td>
</tr>
<tr>
<td>5.5</td>
<td>A Sample of the Profile-Profile Aid</td>
<td>87</td>
</tr>
</tbody>
</table>
CHAPTER I. INTRODUCTION

A document retrieval system is one which is capable of retrieving bibliographic data in response to search requests. The data for each bibliographic item minimally includes author(s), title, and journal (or document) reference data, but for some databases abstracts and other information may be included.

One of the more difficult tasks involved in attempting to retrieve information from such systems is knowing how to adequately specify the search topic. If the keywords or phrases used to specify the search topic do not coincide with those used to represent the same topic in the database, the retrieval results will be poor. If the user can't think of the right words to use for a particular database, what aids are available to help him or her out of this dilemma? This research is oriented toward the development and evaluation of online, interactive aids to assist the user who is confronted with the above problem.

One attractive source of information suitable for interactive aids, which seems to have been ignored by retrieval system designers, is the set of prior search requests which have been queried against the system. In most retrieval environments there are questions which are constantly recurring. This is especially true in a university environment with the annual turnover of students performing searches on recurring term paper topics, etc., and with the high mobility of faculty and staff performing research in related areas.
Consequently, the first two aids developed in this research attempt to make use of the intellectual effort expended by other users who have previously queried the system. One of the aids will enable a new user to automatically locate and display any existing profile\(^1\) which is similar to the one he is developing so that he can make use of the entire profile or any pertinent part thereof. The other aid will help the user to expand his profile by displaying any earlier profile terms (not necessarily from a single profile) which appear to be related to the ones he initially thought of; the relatedness of these terms is determined on the basis of interaction of the various terms in the existing set of profiles.

The above two aids should work well for those cases where a new query resembles an earlier one. However, there will be many cases where a new profile will not resemble any existing profile in whole or in part. For these cases, a third aid will provide assistance by displaying new terms whose potential relatedness to the user's initial search terms is determined on the basis of interaction of the various terms found in a representative set of documents.

The above three aids help the user broaden his search by enabling him to include any necessary synonyms and related terms which he may have overlooked. The fourth aid will help a user\(^1\)

---

\(^1\)The search requests, when coded in a form required by a computerized retrieval system, are usually referred to as "queries", "user interest profiles", or simply as "profiles"; hence, each of these terms as well as simple variations thereof, will be used interchangeably with "search request" in this dissertation.
focus his search, if necessary, by allowing him to make relevance judgments on some minimum number of documents retrieved from a search profile developed at some particular point. The system will respond by suggesting potential terms and appropriate Boolean logic needed to eliminate unwanted documents.

As mentioned earlier, another goal of this research was to evaluate the above aids in as comprehensive a manner as possible. It is of little value to incorporate aids designed to help users of a retrieval system without testing to see if the aids are performing the desired functions. Such an evaluation was performed, but it was not an easy task because of the difficulty of designing a test and evaluation experiment which takes into account the many human performance variables involved.

The organization of the remainder of this document is as follows. Chapter 2 provides some necessary background and perspective for the type of research performed in this dissertation. Chapter 3 deals with the problems of evaluating information retrieval systems and their components, and also discusses the approach used in this research for comparative evaluation of retrieval system output. Chapter 4 discusses the environment in which the research was conducted, while Chapter 5 provides a detailed description of the online interactive profiling aids discussed briefly above. Chapter 6 discusses design considerations, design alternatives, and the design actually used for experimental evaluation of the Aids system and the individual aids. Chapter 7 provides and
discusses the experimental results from the evaluation studies, while Chapter 8 discusses the conclusions to be drawn from the research together with suggestions for future research.
CHAPTER II. DOCUMENT RETRIEVAL SYSTEMS - AN OVERVIEW

As is done in much of the literature, the phrases "document retrieval system" and "information retrieval system" will be used interchangeably in this dissertation to refer to computerized systems capable of retrieving bibliographic reference data in response to user search requests. This chapter provides some background relevant to the research performed in this dissertation on development and evaluation of interactive aids for improved searching of document retrieval systems.

2.1 A Brief History

During the decade of the 1960's, the application of computers to retrieval of bibliographic data began in earnest. Traditionally this information had been, and often still is, retrieved manually through printed indexes and card catalogs. Although edge-notched cards, optical coincidence systems, primitive computer systems, and other mechanized or semimechanized devices were introduced in the 40's and 50's, the big shift toward the development and use of computerized document retrieval systems began in the 1960's.

These early computerized retrieval systems worked in an off-line, batch processing mode, and despite their speed and ability to do new and unique types of searches, they had some inherent disadvantages as pointed out by Lancaster and Fayen (21). First, these systems permitted no browsing, except for that permitted by the printed versions of some of the databases being searched.
Second, the search strategy could not be developed heuristically because of turnaround-time delays; the searcher basically had one chance to conduct a successful search or would have to wait days or weeks to get feedback from a previous trial before making another iteration. Finally, the search had to be delegated to an information specialist who was knowledgeable about details of the query language commands and syntax of the system and database being searched. This delegation often led to poor search results due to a communication breakdown between user and specialist.

To overcome some of the problems inherent with the batch retrieval systems, yet maintain their strengths, the late 1960's saw the advent of experimental online, interactive retrieval systems. By the early 1970's, concomitant with significant advances in computer and telecommunications technology, there was a definite trend both to convert the existing batch systems to online operation and to develop new online systems. In 1973 Lancaster and Fayen (21) discussed 29 major online retrieval systems, some purely experimental, but many fully operational.

The early online systems all contained the following basic features. As the user entered a search term, the system responded by indicating the number of documents indexed (or posted) under the entered term. Through appropriate commands these systems also allowed online display of the retrieved bibliographic records at the terminal, and most permitted the information to be printed off-line on a high speed printer. In addition, although the systems
allowed boolean combinations, search terms in many cases could only
be entered one per line, and then had to be combined later in the
search by using the line numbers representing the retrieved sets
for the individual terms.

The novel or advanced features exhibited by some of the early
systems were soon adopted by the other major systems, and, although
the query language syntax differed to some extent, all major
systems soon had a more advanced set of features. These features
included direct entry of terms in boolean expressions, right
truncation of terms, and display of index terms alphabetically
adjacent to the entered term as a type of browsing aid for
truncation. In many systems the display also included the postings
associated with these alphabetically adjacent index terms (e.g., by the
ORBIT command "NEIGHBOR" (46)). Today, the major commercially
available online information retrieval systems provide all of the
above features and more. Depending on which of the many available
databases are searched, searches can be performed selectively on
any available fields such as title, index terms, or abstracts. For
those databases with whole or partial text fields (such as an
abstract), in addition to boolean connectives, commands are avail­
able to require the words to be in the same paragraph, in the
same sentence, within a given number of words of each other, or
adjacent to each other in the text.

Despite the full set of features described above, few actual
user aids are provided by the major commercial retrieval systems.
Why such aids are useful and the types of aids available either in operational systems or experimental systems are discussed later in this chapter.

2.2 Who Should Interact with the System

Before discussing user aids in more detail, the audience to which such aids are directed must be discussed. As mentioned above, most batch retrieval systems of the 1960's required the presence of an intermediary, usually called an information specialist, to run a search. The job of the information specialist working in a batch environment is to determine the needs of the person requesting the search (the end user), usually through an interview, and then translate the search request into the query language of the system on which the search is to be run.

A communication breakdown between user and intermediary can lead to poor search results. This breakdown can be caused by the user's inability to accurately describe his needs, the intermediary's inability to interpret the user's needs, or, more likely, a combination of the two. This problem has been discussed by many authors, but a good discussion of this problem and how it fits into the overall problem of retrieval system accuracy is provided by King and Bryant (19).

With the advent of online interactive systems, many feel that one solution to the above communication problem is to simply eliminate the intermediary and allow the end user to conduct his
own search. Such sentiment is expressed by Lancaster and Fayen (21) who state:

"Ultimately, ... on-line systems should be capable of being used in a nondelegated search mode. That is, the practitioners in a field should be able to undertake, productively, their own literature searches without the interposition of an information specialist. The problems of misinterpretation and miscommunication are thereby avoided."

However, because the current generation of end users are unskilled in online searching the need for an intermediary still exists despite the communication problems mentioned earlier.

An argument for retaining the intermediary is provided by Williams (57) who found that a search by an intermediary provided an effective search at the lowest cost, but that the most effective search is one where both specialist and user participate. This allows the two to exploit the user's subject knowledge as well as the intermediary's knowledge of and experience with the search system.

Assuming for the moment that it is in fact desirable to eliminate the intermediary and put the end user directly online, is it feasible? There are several major online retrieval systems, each with its own query language. In addition, for a given retrieval system, each different database has different characteristics: some have been indexed manually and can be searched using controlled index terms while others can be searched only by using free-text title or abstract words. Some contain abstracts, others do not. The number and types of searchable fields differ from database to database. Due to this lack of
uniformity, it seems that the casual user could simply not keep up with the necessary technical aspects of the databases and retrieval systems he might have occasion to use.

Despite these problems, many feel that eventually the end user will be the primary online searcher, with the information specialist being used in only "top of the line" searches. This sentiment is expressed by Meadow (28) who draws a parallel between bibliographic searching and computer programming: Meadow feels that the present online command languages are analogous to the early assembler languages of computer programming. Only a relatively few people have knowledge of these languages (i.e. the intermediaries), and hence the end user must deal with the online system through the trained intermediary. However, just as assembler languages in programming gave way to higher level languages which opened the door to amateur programmers, Meadow feels that the next generation of search systems' command language will allow the end user to handle routine information requests on his own, needing the help of the specialist only when he gets past the elementary level.

Williams (54) has a similar sentiment. She feels that the variability in systems and databases discussed above presents barriers and hindrances to the end user trying to gain familiarity with online systems. She feels that the end user will be able to go online as user-oriented transparent systems are developed. Such systems would be user friendly and would automatically convert from the user friendly language to the necessary system language, thereby alleviating the user from the need to understand differences
between systems, databases, command languages, vocabularies, and logon protocols. Initial work in this area has already begun. For example, there has been work at MIT and Drexel on a common command language (See, for example, Marcus (26,27) and Meadow (29)). Further work on components of a transparent retrieval system has been done at Illinois on an automatic database selector (56) and at Battelle on a system to automatically switch from one index vocabulary to another (32,33,34).

Therefore, although we may see a shift toward end user searching in the near future, it will likely be some time before the majority of online searches are performed without the assistance of an intermediary.

2.3 Online Search Aids - Background

For centuries there has been a need to index or classify documents. This is necessitated by the fact that information must somehow be accessible by topic or area of interest rather than some arbitrary ordering (such as chronological ordering) which would make information available only by some type of sequential scan through the literature. Thus, we have had, among others, the Dewey Decimal System (9) the Universal Decimal Classification (4), and the Library of Congress Codes (38) to classify books. Individual disciplines (such as Chemistry, Biology, and Medicine, etc.) have categorized the literature of interest by using various schemes, most by using hierarchial classification or subject
headings to divide the literature into appropriate classes. Unfortunately, because these schemes were used for filing purposes, which necessitated that an item be assigned to only one class, the orderings proved advantageous only for those users whose views of the discipline coincided with that of the classification scheme.

With the advent of computerized search systems, and the ability of the computer to quickly combine various sets of classified documents in new ways, other possibilities of indexing and retrieving information became feasible. As many classes as applicable could be assigned to a document, and then the retrieval system could be used to intersect and otherwise combine classes to provide greater retrieval flexibility. Further, the practice of keyword indexing became prevalent. Instead of assigning documents to a class, keywords or phrases from a fixed list were assigned to describe the document. This scheme had the advantage that as a field of knowledge grew, new keywords could easily be added to the allowed list without disrupting the existing scheme.

Fixed keywords also had the advantage of vocabulary control. A thesaurus could be set up to not only list the allowable index terms, but to list synonyms of the allowable terms and indicate which terms should be used in indexing. Thus the user searching the file could consult the thesaurus to find out that if he is interested in information about cars, he should look up AUTOS instead of CAR, CARS, AUTO, AUTOMOBILE, or AUTOMOBILES. Again, the intellectual effort invested by the indexer aided the user during retrieval.
The process of vocabulary control of any type, does have a few disadvantages. In many cases the classification used, or even the groupings forced by the fixed keywords, does not meet the needs of the end user. For this type of user, the search is often awkward and may retrieve a large amount of irrelevant information. Further, the intellectual effort invested by the indexer is expensive, involving a cost of both time and money.

In order to avoid the above problems, many databases have eliminated the manual indexing process, and these databases are searched by using the words contained in the title and abstract (if one is available). While this practice has solved some problems, it has created others. In order to adequately search these free-text databases, the user is forced to include in his search strategy all the various terms and phrases which can be used to describe the concept of interest. Trying to think of all such terms and phrases is a difficult task not only for the intermediary who is not a specialist in the area covered by the search, but also for the subject specialist.

This problem has been discussed extensively in the literature. For example, in discussing the traits of a good intermediary, Dolan and Kremin (12) list the ability to think in synonyms as one requirement. These researchers feel that this trait is equally important for controlled and uncontrolled vocabulary databases, and further state that "It is impossible to rely on the users to suggest all synonyms, because often they use only preferred terms, or are even unaware that any synonyms exist." When discussing search strategy,
Quinn (39) lists as one guideline the use of an online "synonym table," or at least a printed synonym dictionary.

What has been done to help the user in this area of expanding a search to adequately cover a concept? Some aids are available, but more help should be provided. Many databases use thesauri, as mentioned above, but the cost of thesaurus preparation and the continued indexing costs are major drawbacks, and hence thesauri do not exist for many databases. Further, a general thesaurus or one prepared for another database is often of limited value to the database being searched.

Much work has been performed in an attempt to make use of information already available to provide online assistance to the user. Most of the aids have been confined to experimental or small scale retrieval systems. These aids are discussed in the next several sections. Additional aids proposed in this research are discussed in a later chapter, and a system based on these and several other user aids is also described and evaluated.

2.4 Current Online Search Aids

To be of maximum benefit, a search aid should be available online. Although thesauri and synonym dictionaries are sometimes available in printed form, these are primarily useful in presearch preparation. Once the user or intermediary sits down at the terminal and begins an online session, such printed aids become bulky and awkward to use.
Currently, the major commercial information retrieval systems contain only a few aids to help the user expand his search online. One such aid is the display of terms in the index which are alphabetically adjacent to the entered term, as discussed earlier. This aid can at least help the user include many or all of the various inflectional forms of a word, help prevent over-truncation, and often help the user uncover variant spellings. As a second aid, online thesauri are being made available for a few databases. As mentioned earlier, an online thesaurus should prove superior to its printed counterpart once the user has started an online session. (It also obviates the need to purchase a printed thesaurus for a database which is used only occasionally.) Finally, a relatively new capability has been introduced recently to allow a user to receive postings from several databases at once in order to facilitate database selection (e.g. ORBIT's CROSSFILE search, and BRS's CROS file). These aids, however, only partially meet the need for concept coverage.

2.5 Association Measures and Their Computation

In addition to the above aids, many others have been suggested, and several of these have been implemented in small or experimental retrieval systems. Since several of the experimental aids have been based on association measure techniques, it is advisable to examine the basic theory underlying these techniques before discussing the aids developed using the theory.
Association measures or correlation measures have long been utilized in the field of statistics to assign a similarity value to two objects. These measures have been used in the information field for over two decades in a variety of contexts. One example is the use of an association measure to assign a value of similarity between a search request and a document based on the terms common to each other. Another example is the use of these measures to assign a value to the similarity between a pair of search terms based on the number of documents in which each occurs.

A simple example which illustrates the use of association measures in information retrieval is presented below (based on Salton (42)). For example, we will assume a universe of three documents (designated D1, D2, and D3) which are indexed by eight unique index terms (designated T1 – T8). We will further assume that

- Document D1 is indexed by T1, T2, T3, T7, T8,
- Document D2 is indexed by T1, T2, T3,
- Document D3 is indexed by T1, T2, T3, T4, T5, T6, T7, T8.

Each document can then be represented as a vector of dimension t, where t represents the total number of unique index terms (t=8 in the example). Thus, we can represent document i (Di) by the vector

$$Di = (w_{i1}, w_{i2}, \ldots, w_{it})$$

where \(w_{ij}\) is 1 if term \(j\) is used to index (or otherwise represent) document \(i\), and \(w_{ij}\) is 0 if term \(j\) is not used to index document \(i\).
The three documents described above can be represented by vectors as follows:

\[ D_1 = (1,1,1,0,0,0,1,1) \]
\[ D_2 = (1,1,1,0,0,0,0,0) \]
\[ D_3 = (1,1,1,1,1,1,1,1) \]

By a simple extension, a search request can also be represented using the vector notation. For example, if search request \( Q_1 \) contains terms \( T_1, T_2, \) and \( T_3 \), then \( Q_1 \) can be written

\[ Q_1 = (1,1,1,0,0,0,0,0) \]

The general case is

\[ Q_1 = (v_{11}, v_{12}, \ldots, v_{1t}) \]

where \( v_{ij} \) is 1 if term \( j \) is contained in search request \( i \), and 0 otherwise.

Many correlation measures exist which can be used to measure the similarity between a request and a document. One such measure is the vector product which can be represented by

\[ S_V (D_i, Q_j) = \sum_{k=1}^{t} w_{ik} v_{jk} \quad (2.1) \]

where \( S_V (D_i, Q_j) \) represents the similarity between document \( i \) and search request \( j \) as measured by the vector formula (i.e. the formula in Equation 2.1), and \( t, w_{ik}, \) and \( v_{jk} \) are defined above. A sample calculation of this measure is presented in Figure 2.1(a).

Although the vector product measure is appealing because of its simplicity, this measure suffers from the fact that only the co-occurrence of terms is taken into account, and not the individual occurrences or the total number of terms used in the request or
Document and Query Vectors

\[ \mathbf{D} = (1,1,1,0,0,0,1,1) \]
\[ \mathbf{D} = (1,1,1,0,0,0,0,0) \]
\[ \mathbf{D} = (1,1,1,1,1,1,1,1) \]
\[ \mathbf{Q} = (1,1,1,0,0,0,0,0) \]

(a) \[ S_V (D1, Q1) = \sum_{k=1}^{8} w_{1k} v_{1k} \] (Equation 2.1)

\[ = 1 + 1 + 1 + 0 + 0 + 0 + 0 + 0 \]
\[ = 3 \]
\[ S_V (D2, Q1) = \sum_{k=1}^{8} w_{2k} v_{1k} = 3 \]
\[ S_V (D3, Q1) = \sum_{k=1}^{8} w_{3k} v_{1k} = 3 \]

(b) \[ S_T (D1, Q1) = \frac{\sum_{k=1}^{8} w_{1k} v_{1k}}{\sum_{k=1}^{8} w_{1k} + \sum_{k=1}^{8} v_{1k} - \sum_{k=1}^{8} w_{1k} v_{1k}} \] (Equation 2.2)

\[ = \frac{3}{5+3-3} = \frac{3}{5} = .60 \]
\[ S_T (D2, Q1) = \frac{3}{3+3-3} = \frac{3}{5} = 1.00 \]
\[ S_T (D3, Q1) = \frac{3}{8+3-3} = \frac{3}{8} = .375 \]

Figure 2.1. Association Measure Computation Examples
(a) Vector Product Measure (an unnormalized measure)
(b) Tanimoto Measure (a normalized measure)
document. Thus, in the present example, request Q1 is rated equally similar to the three documents, but it seems to be most closely related to document D2. This fact should be reflected by the association measure.

Other measures have been developed which overcome this problem by using a form of normalization to account for the total number of terms contained in a document or profile. One such measure, introduced by Tanimoto (51) is shown in Equation 2.2.

$$S_T (D, Q) = \frac{\sum_{k=1}^{t} w_{1k} v_{jk}}{\sum_{k=1}^{t} w_{1k} + \sum_{k=1}^{t} v_{jk} - \sum_{k=1}^{t} w_{1k} v_{jk}}$$

In this equation, $S_T$ represents the similarity between two objects as measured by the Tanimoto formula; the other symbols used have been defined above. As shown by the sample calculations in Figure 2.1(b) this measure produces association values which are intuitively more appealing than those produced by the vector product measure.

Many other measures have been proposed and studied. A good discussion of several association measures can be found in Salton (42). For an interesting article which provides a criterion for comparing the various association measures, the reader is referred to Jones and Curtice (18).

The association techniques can be further applied to provide a measure of association between two terms or between two documents. For this, an extended form of notation is helpful and is shown in Figure 2.2. Here it is shown that the document vectors can be
### Document Term Matrix

<table>
<thead>
<tr>
<th></th>
<th>T1</th>
<th>T2</th>
<th>T3</th>
<th>T4</th>
<th>T5</th>
<th>T6</th>
<th>T7</th>
<th>T8</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>D2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>D3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

\[ c = \begin{bmatrix}
  1 \\
  1 \\
  1 \\
  1 \\
  0 \\
  0 \\
  0 \\
  0 \\
\end{bmatrix} \]

\[ Q_1 = \begin{bmatrix}
  1 \\
  1 \\
  1 \\
  0 \\
  0 \\
  0 \\
  0 \\
  0 \\
\end{bmatrix} \]

\[ R_1 = C \cdot Q_1 \quad \text{(Equation 2.3)} \]

\[ R_1 = \begin{bmatrix}
  3 \\
  3 \\
  3 \\
\end{bmatrix} \]

Figure 2.2. Associative Retrieval - An Example
combined to form a document-term matrix, \( C \), with \( d \) rows and \( t \) columns, where \( d \) is the total number of documents in the collection, and \( t \) the number of terms (\( d=3 \) and \( t=8 \) in the above example). Then, for example, we can view retrieval by keyword matching as a simple vector multiplication of the form

\[
R_i = C \cdot Q_i
\]  

(2.3)

In Equation 2.3, \( C \) is the document-term matrix discussed above, \( Q_i \) is the query vector (written as a column vector as shown in Figure 2.2), and \( R_i \) is the retrieval vector of dimension \( d \). In this technique, note that each element in vector \( R_i \) represents the binary weight assigned to document \( D_j \) by the system. Again, it can be seen that this type of matching corresponds to the first association measure presented in Figure 2.1(a).

Using matrix \( C \) in Figure 2.1, we can also apply an association measure to any two columns to obtain a measure of similarity between the corresponding two terms. Likewise, we can apply an association measure to any two rows to obtain a measure of similarity between the corresponding documents. Thus, using the Tanimoto measure as the matrix operator, we can compute \( C^T \cdot C \) (where \( C^T \) is the transpose of \( C \)), to obtain a \( t \times t \) term-term similarity matrix \( B \) as illustrated in Figure 2.3. Similarly, we can compute \( C \cdot C^T \) to yield a \( d \times d \) document-document association matrix \( A \) as illustrated in Figure 2.4.
**Figure 2.3.** Term-Term Association Matrix (B) Computed from Document-Term Matrix (C) in Figure 2.2 Using the Tanimoto Association Measure

<table>
<thead>
<tr>
<th></th>
<th>T1</th>
<th>T2</th>
<th>T3</th>
<th>T4</th>
<th>T5</th>
<th>T6</th>
<th>T7</th>
<th>T8</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>.33</td>
<td>.33</td>
<td>.67</td>
<td>.67</td>
<td></td>
</tr>
<tr>
<td>T2</td>
<td>1.0</td>
<td>1.0</td>
<td>.33</td>
<td>.33</td>
<td>.33</td>
<td>.67</td>
<td>.67</td>
<td></td>
</tr>
<tr>
<td>T3</td>
<td>1.0</td>
<td>.33</td>
<td>.33</td>
<td>.33</td>
<td>.67</td>
<td>.67</td>
<td></td>
<td></td>
</tr>
<tr>
<td>T4</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>.50</td>
<td>.50</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T5</td>
<td>1.0</td>
<td>1.0</td>
<td>.50</td>
<td>.50</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T6</td>
<td>1.0</td>
<td>.50</td>
<td>.50</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T7</td>
<td></td>
<td></td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.0</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 2.4.** Document-Document Association Matrix (A) Computed from Document-Term Matrix (C) in Figure 2.2. Using the Tanimoto Association Measure

<table>
<thead>
<tr>
<th></th>
<th>D1</th>
<th>D2</th>
<th>D3</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1</td>
<td>1.0</td>
<td>.60</td>
<td>.625</td>
</tr>
<tr>
<td>D2</td>
<td>1.0</td>
<td>.375</td>
<td></td>
</tr>
<tr>
<td>D3</td>
<td>1.0</td>
<td></td>
<td>1.0</td>
</tr>
</tbody>
</table>
In the general case, the matrix C can be represented as shown in Figure 2.5. To apply the Tanimoto measure to a pair of term vectors to compute one element in the term-term matrix, Equation 2.4 should be used.

\[ S_T(T_i, T_j) = \frac{\sum_{k=1}^{d} w_{ki} w_{kj}}{\sum_{k=1}^{d} w_{ki} + \sum_{k=1}^{d} w_{kj} - \sum_{k=1}^{d} w_{ki} w_{kj}} \]  

In this equation, \( S_T(T_i, T_j) \) represents the measure of similarity between term \( T_i \) and term \( T_j \), and the other terms have been defined previously.

![Figure 2.5. The Document-Term Matrix (General Case)](image)

Similarly, Equation 2.5 can be used to apply the Tanimoto measure to a pair of document vectors to compute one element in the document-document matrix.
In this case, \((D_i, D_j)\) represents the measure of similarity between document \(D_i\) and document \(D_j\). Again, the other terms have been previously defined.

All of the above techniques can be combined into a more sophisticated retrieval operation, known as linear associative retrieval, first reported by Giuliano and Jones (16). This technique is also described in Salton (42). Specifically, if we denote the document-document association matrix by \(A\), and the term-term association matrix by \(B\), we can view retrieval as

\[
R_i = A \cdot C \cdot B \cdot Q_i \quad (2.6)
\]

In this equation, \(B \cdot Q_i\) produces a \(t\)-dimensional column vector which can be considered to be a request vector expanded by including terms highly associated to those in the initial query. The product \(C \cdot D \cdot Q_i\) then produces an initial retrieval vector (of dimension \(d\)) which represents retrieval based on the expanded query. Finally, multiplying this result by the \(A\) matrix adds to the retrieved set any documents not in the set which are highly associated to those which are in the set. This final result set, then, represents the outcome of retrieval enhanced both by associated terms and associated documents.

Although linear associative retrieval may be viewed as a user aid, it is somewhat different than the kinds of aids under discussion, so it has been included mainly as a further example of the
association theory. The aids of interest which have been based on this theory are described in the next section.

2.6 Experimental Aids Based on Term-Term and Document-Document Associations

In addition to the aids discussed in Section 2.4, many other aids have been suggested, and several of these have been implemented in small or experimental retrieval systems. The major aids of this type will be discussed in this section and the following section.

The first type of aid discussed here is based on the association measure techniques discussed in the last section. These techniques have been studied since the early 1960's, and they have been applied experimentally in various settings. They have been utilized for automatic document classification, indexing, thesaurus building, and as a means for generating synonym and related term lists. The basic theory behind the use of association measures was discussed in the last section, but the various uses to which these techniques have been applied are discussed below.

Basically, the idea of statistically related terms is derived from the belief that if two terms co-occur in a large number of documents, the terms are related. Much work was done with these term-term associations (see for example Doyle (14) and Stiles (50)) and the work met with some success. One problem which seemed to arise, however, is that if the associations were based on co-occurrences of terms in titles, the resulting highly associated pairs tended to exhibit a contiguity relationship - i.e. they tended to occur in the same phrase, one word often used to modify the other, etc.
In order to try to improve the associations to represent synonymy, attempts were made to use second order term-term associations (Libbey (24), Lewis (23)). In this approach, words were considered to be related if they were associated at the first order level to a similar list of words. The theory espoused here is that words which exhibit first order relationships are contiguous words, while synonymous terms have similar first-order contexts. The similarity of the first order context is measured by the second order association value. While this technique worked as planned for a few word pairs, in general, too many unrelated pairs were found to make second order associations a viable tool.

While the above techniques and many others similar to them have been described in the literature, and several have been implemented in small experimental retrieval systems, these techniques have not found their way into the large operational retrieval systems. This is due primarily to the use of matrices for the computations as discussed in the last section. This matrix technique is neither efficient nor cost effective for use with large systems due to the resulting size of the matrices. However, two recent applications may prove to be useful with large systems.

The first of these is the Associative Interactive Dictionary (AID) developed by Doszkocs (13). Doszkocs works only with the retrieved document set and single term occurrence frequencies from the entire document collection. Since the retrieved document set is relatively small, and the collection frequencies are available as part of the inverted index, the quantities necessary for
computation can be located relatively quickly. Further, a computationally simple formula is used to compute relatedness values so that this calculation can be performed in an acceptably short time. Doszkocs' relatedness value $S_D$ is then computed as follows:

$$S_D = \frac{f_o - f_e}{f_o}$$  \hspace{1cm} (2.7)

where $f_o$ is the observed frequency of the term in the retrieved set, and

$$f_e = \frac{nm}{d}$$  \hspace{1cm} (2.8)

is the expected number of document occurrences in the retrieved set. The value for $f_e$ is computed by multiplying the total number of documents in the retrieved set ($n$) by the number of documents in which the term occurs in the entire document collection ($m$), and dividing this result by the total number of documents in the file ($d$). If a term occurs significantly more often than expected in the retrieved set, it is assumed to be related to the term which was used to generate the retrieved set.

A further feature mentioned by Doszkocs is that of an internal hash table. This "compressed in-core hash table" allows rapid lookup of the document frequency information for the entire collection which is necessary to compute the relatedness measure described above. A prototype version of the AID is operational on TOXLINE, a system for querying a large file of over 400,000 journal articles on Toxicology.

The second recent application is described by Preece (37). Preece has developed an Online Associative Query System (OAQS) which
is capable of "relevance feedback, thesaurus and statistical query expansion, Boolean and best-match searching, and retention of associations based on previous search experience". The uniqueness of this system stems from the fact that the document-term matrix (see Figure 2.5) is viewed by Preece as a network as shown in Figure 2.6. The links in the figure indicate the use of the term represented by a node $T_i$ in the document represented by a node $D_i$. This structure allows the computation of various features by traversing the links and allowing the weights assigned to the various nodes to "diffuse" through the network. A pilot version of this interesting system has been implemented on a DEC-10, and tested on small scale files.

![Figure 2.6. A Document-Term Network](image)

2.7 Profile Retrieval Aids

Two profile retrieval aids have been discussed in the literature and will be described in this section. Neither uses the associative techniques applied in this research to isolate similar profiles, but both have as their objective the retrieval
of stored profiles which would assist the user in constructing his profile.

The first system which included this aid was the Syracuse University Psychological Abstracts Retrieval System (SUPARS). This experimental, online retrieval system was implemented in the early 1970's at Syracuse University to search a file of psychological abstracts, and was described in some detail by Lancaster and Fayen (21). The feature of interest allowed the retrieval of stored profiles in two forms. First a search term could be entered followed by the command "SHOW PROFILES", which would then display all stored profiles which contained the entered term. In the second form, the search term was followed by the command "SHOW WORDS" which would display all words which occurred in a stored profile which also contained the entered term. The new words would be displayed in descending order of the number of such profiles in which each occurred.

SUPARS seems to be the first system which provided this type of aid. Although such an aid is extremely interesting, it is questionable whether one search term is enough to provide good discrimination for selecting similar profiles. Even though the aid itself has been described, no information concerning its usefulness in the operational system has been provided.

More recently, a second system has included a profile retrieval aid. The Foreign Technology Division (FTD) of the United States Air Force has provided this capability for a Profile System used with
the CIRC II retrieval system. This feature is described in FTD's "The Profile System" (15), and basically allows the user to search a file of stored profiles as he would any other database. The search request may include such features as truncation and boolean connectives, and seems to be a better way to provide similar profile retrieval than that provided by SUPARS. Unfortunately, again no results of evaluation of this aid have been published.

2.8 The SMART System's Relevance Feedback

No discussion of retrieval aids would be complete without mentioning the work Salton and his co-workers have performed using the SMART retrieval system (42,43,44). Using the SMART system, these researchers have studied a wide variety of automatic retrieval system features using relatively small experimental databases. Automatic thesaurus construction, association techniques, retrieval strategies, system evaluation, and clustering are only a few of the topics which have been included.

Of particular interest in the context of a retrieval aid is the technique of relevance feedback used with the SMART system. After an initial query by the user, the system displays the retrieved documents and the user indicates whether each is relevant or nonrelevant to his interests. The system then automatically adjusts the weights of the query to increase the weights of those terms which occur predominantly in the relevant group of retrieved documents. The weights of the other terms are lowered (this is much like the SDI profile adjustment discussed in Section 4.1.2).
While this technique can be used to adjust weighted queries, it is not appropriate for dealing with systems using unweighted, boolean search queries. The type of relevance feedback discussed in Section 5.5 should prove more useful for this type of system.
CHAPTER III. THE EVALUATION OF RETRIEVAL SYSTEMS AND THEIR COMPONENTS

When new features are introduced into information retrieval systems, the immediate question which arises is, "How well do they work?" To answer this question, some type of evaluation must be performed on the proposed feature. This often simply entails qualitative checking to make sure the feature does not appreciably degrade response time, and qualitative checking to see whether users tend to "like" the feature. However, when systems and features are evaluated more carefully it is desirable to have some quantitative or semi-quantitative measures for this purpose. Several such measures used by previous workers have emerged as fairly standard measures of retrieval system performance despite some difficulties encountered with some of them. This chapter discusses some of the more common measures together with their applicability toward the approach used in this study.

3.1 Precision, Recall and Other Measures

The measures commonly used to evaluate retrieval system performance can best be described with the aid of Table 3.1. As shown in this contingency table, the collection of documents searched by a retrieval system can be partitioned in several ways for a particular user and a particular search request. One way of partitioning is on the basis of those documents retrieved (a+b) and those not retrieved (c+d) as answers to a particular search request. Another way of partitioning is on the basis of those documents in the collection deemed to be relevant to a particular user (a+c) and
those deemed to be not relevant \((b+d)\). (The notion of relevance is discussed below.) The set represented by \(a\) in Table 3.1 is that group of documents retrieved by the system which are also relevant to the user. The sets represented by \(b, c,\) and \(d\) in the table are defined analogously.

Table 3.1

<table>
<thead>
<tr>
<th>Relevant</th>
<th>Not Relevant</th>
<th>Totals</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retrieved</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(a)</td>
<td>(b)</td>
<td>(a+b)</td>
</tr>
<tr>
<td>Not Retrieved</td>
<td>(c)</td>
<td>(d)</td>
</tr>
<tr>
<td>Totals</td>
<td>(a+c)</td>
<td>(b+d)</td>
</tr>
</tbody>
</table>

Using the symbols defined in Table 3.1, a number of retrieval performance measures can be defined such as those shown in Table 3.2. These measures and the symbols used to represent them can be briefly described as follows. Recall \((R)\) is a measure of the fraction of relevant documents in the file which are actually retrieved. This indicates how effectively the search is finding all relevant material in the file. Precision \((P)\) is a measure of the fraction of retrieved documents which are relevant, and indicates the efficiency with which the relevant documents are retrieved. Fallout \((F)\), which measures the fraction of nonrelevant documents retrieved, indicates how well the nonrelevant are rejected as a function of the size of
the document collection. Finally, Generality (G), the proportion of
documents in the collection which are relevant to the query, is a
measure of the density of relevant documents in the entire collection.
Precision and Recall have probably been the most widely used
measures, but the above four measures are not independent of each
other, and the impact of this on the choice of measure used for any
particular study has been discussed by Salton (44).

Table 3.2

Retrieval Evaluation Measures

<table>
<thead>
<tr>
<th>Measure</th>
<th>Symbol</th>
<th>Formula^z</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recall</td>
<td>R</td>
<td>( \frac{a}{a+c} )</td>
</tr>
<tr>
<td>Precision</td>
<td>P</td>
<td>( \frac{a}{a+b} )</td>
</tr>
<tr>
<td>Fallout</td>
<td>F</td>
<td>( \frac{b}{b+d} )</td>
</tr>
<tr>
<td>Generality</td>
<td>G</td>
<td>( \frac{a+c}{a+b+c+d} )</td>
</tr>
</tbody>
</table>

^zThese formulas, which represent fractions, are also frequently
represented as percentages by multiplying the fraction by 100.

There are several other measures which are of potential interest
due to their user orientation. One is the novelty ratio, the number
of relevant items retrieved of which the user had not been aware
divided by the number of relevant items retrieved. Another is the
coverage ratio, the number of relevant items retrieved which were
already known to the user divided by the total number of relevant
items already known to the user. A final measure in this category is relative recall which is the ratio of the number of relevant documents retrieved to the number of relevant documents the user wished to retrieve. A good review of the above measures and several others is provided by both Salton (44) and Lancaster (20).

Despite the abundance of search performance measures, in most cases either the Recall and Precision pair or the Recall and Fallout pair is used to measure retrieval system performance. The Recall/Precision measure is felt to be user oriented, with Precision measuring the efficiency with which the relevant are retrieved, while Recall/Fallout is felt to be systems oriented, with Fallout indicating how well the nonrelevant are rejected as a function of collection size.

Despite the widespread use of the above pairs, especially Recall/Precision, there are problems with the use of Recall. In order to accurately measure Recall, every document in the collection would have to be examined for relevance. While this has been done by some experimenters for small collections (100-200 documents), it is at best impractical for all but the smallest document collections. Therefore, in the current research, the number of relevant retrieved documents and the precision of the search (the quantities represented by a in Table 3.1 and P in Table 3.2 respectively) will be used to compare search performance in this dissertation as discussed more fully both in the next section and in Chapter 6. (However, the symbol \( r \) will be used rather than \( a \) to represent the number of retrieved items which are relevant.)
Before proceeding, a note of caution on the use of relevance is in order. There has been much discussion in the literature concerning the inherent problems associated with relevance judgements (see for example Saracevic (45)). Consequently, any problems associated with relevance judgements should be accounted for by the overall experimental design presented in detail in Chapter 6.

3.2 Problems Associated with Retrieval System Comparison

Even if the issue of relevance judgement is ignored, a problem still remains when comparing retrieval performance of two systems based on the number of relevant items retrieved \( r \) and the precision of the search \( P \). The problem stems from the individuality of each user and each question searched on the system. Due to these individual differences, we cannot compare two different queries where \( r=3 \) and \( P=30\% \) for one, and \( r=2 \) and \( P=50\% \) for the other. Either may be an excellent search, a good search, or a poor search depending on the nature of the system, the nature of the question, and the needs of the user. Therefore, it makes no sense to compare two different systems using two different questions, unless we can perform an extremely large number of searches on each system and then compare the systems by using average figures. Even then, the results might be questionable.

Because of this, it is necessary to compare systems by using the same search request on each system and then comparing the results for this search request (see O'Donahue (35) for example). Even this, however, presents some problems, one of which will be discussed now, while others will be discussed in Section 6.1. Due to the dual
nature of the $r/P$ measure and other similar measures, some cases present clear choices as to the superior search, but others do not. For example, a search in which $r=6$ and $P=75\%$ is clearly superior to a search of the same question on the same database with $r=2$ and $P=25\%$ since it is superior in each measure. It is also an obvious choice if a search is superior in one measure and equivalent in the other (e.g., $r=6$, $P=67\%$ better than $r=4$, $P=67\%$ or $r=4$, $P=100\%$ better than $r=4$, $P=67\%$). However, it is not an easy choice if one search is superior in one measure, but inferior in the other (e.g. is $r=6, P=67\%$ better than $r=4, P=100\%$?). This is a judgement best left to the individual user. However, a general scheme for combining these two measures on a single scale is discussed in Section 3.3, which is used as a basis for a proposed utility measure (Section 3.4) to model value judgements that would be made for such comparisons. The actual judgements made in the evaluation for the current research (Table 7.1) are then used to test the validity of the proposed model (Section 7.1.7).

3.3 A Single Scale for Comparing Retrieval System Output

One of the most interesting measures for evaluating a set of retrieved documents was presented by Cooper (7,8) who described a measure of utility (i.e., usefulness) for a search. He starts by making the following assumptions.

IA: All nonrelevant documents have the same average negative utility (this average is the same across different systems).

IB: Let the utility in IA = -1 (an arbitrary convention)

II: All relevant documents have the same average positive utility (probably somewhat greater than 1).
III A: The user examines all retrieved documents in sequence.

III B: The user will stop a search exactly when he examines t nonrelevant documents ('Frustration Point' Stopping Rule).

III C: The user will stop a search exactly when he examines y relevant documents ('Satisfaction Point' Stopping Rule).

IV: The utility of a document is independent of the order in which the retrieved documents are examined.

If we let k = the utility of a relevant document (Assumption II), then if we assume IIIA,

\[ U = kr - r \]  \hspace{1cm} (3.1)

where U is a measure of the utility of the search, r is the number of relevant documents retrieved, and \( \bar{r} \) is the number of nonrelevant documents retrieved (r and \( \bar{r} \) correspond to a and b, respectively, in Table 3.1). Equation 3.1 can also be expressed as

\[ U = knP - n(1-P) \]  \hspace{1cm} (3.2)

where P is precision, and \( n = r + \bar{r} \) = the total number of documents retrieved.

If we replace Assumption IIIA with Assumption IIIB, then Equation 3.1 becomes

\[ U = ks - t \]  \hspace{1cm} (3.3)

where s is the number of relevant documents seen before the t nonrelevant documents are examined, and U and k are as above.

Similarly, if we assume IIIC rather than either IIIA or IIIB, the utility of the search can be represented as

\[ U = ky - z \]  \hspace{1cm} (3.4)

where z is the number of nonrelevant documents seen before the y relevant documents are examined. (U and k are again defined as above.)
Cooper's ideas seem quite reasonable for the most part. However, the utility of a relevant document \((k)\), although assumed constant for a given user, will vary from user to user. Further, rather than being constant, the utility of a relevant document is probably a type of step function, with the utility being higher for the first few relevant documents, and then decreasing in steps as more relevant documents are found. Note, however, that the function would not necessarily decrease as each new relevant document is examined; for a particular range of relevant documents the function may remain constant.

3.4 Proposed Model for Comparative Evaluation of Retrieval Outputs

Whether or not the reader agrees with Cooper's assumptions and his measure, the above discussion introduces the basic problem which must be addressed in developing a single measurement scale for evaluating retrieval system output: such a scale must be developed for the \(r/P\) space, or alternatively, the \(R/P\) space. (These spaces are similar, but not identical\(^a\).) This scale would be an ordinal scale of measurement (see, for example, Stevens (49) for a discussion of scales of measurement). Further, the ordinal scale does not exist a priori (as it would, for example, if we ranked the runners in a race by their order of finish), but must be determined for each individual user by some type of utility measurement. The general theory of utility, upon which Cooper's ideas are also based, had

---

\(^a\)As discussed in Section 3.1, \(r\) is the number of retrieved relevant items (equivalent to \(a\) in Table 3.1) and \(R\) is recall as defined in Table 3.2.
its roots in the field of economics. (Morris (31) provides a good introduction to basic utility theory.)

In general, a weak ordering must be imposed on either the r/P or R/P space. (We will use the R/P space initially for the sake of simplicity.) A weak ordering exists if for points A and B in the space, one and only one of the following holds:

A ≥ B, A = B, or A < B

Also, transitivity must hold (i.e. A ≤ B and B ≤ C ⇒ A ≤ C). In all cases (except possibly boundary cases), if A has higher precision and recall than B, then A>B. However, since precision and recall usually vary inversely (see Lancaster (20), pp. 111-117), how do we divide the space where the recall for A is higher than that for B, but the precision lower?

This problem is shown graphically in Figure 3.1. Assume that the output from a search is evaluated and found to be at a particular point in the R/P space (say point A). Also assume that the "ideal" search is represented by point B (P=100%, R=100%). The space must be ordered so that we can draw a line (or possibly an area), containing point A, which has the characteristic that all points on the line (or in the area) represent search results which are considered equivalent to results represented by point A (for a given question and a given database). Further, the line (or area) containing point A divides the R/P space into two areas. Any point in one area would represent improved search results, while any point in the other area would represent poorer search results. Thus, as
Figure 3.1. The Problem of Dividing the R/P Space

Figure 3.2. An Example of Dividing the R/P Space
illustrated in Figure 3.2, if the line CD represents points which are equivalent, then any point above and to the right of the curved line CD represents search results superior to those represented by A, while any point on the other side of the line represents search results inferior to those represented by A.

In general, if the idea (presented in Section 3.2) that a superior search is one which is superior in either the P or the R measure and no worse in the other is accepted, then the quadrant labeled "Quad 1" in Figure 3.1 represents an area which is clearly superior, while the quadrant labeled "Quad 3" in Figure 3.1 represents an area which is clearly inferior. The question, then, is how to divide the quadrants labeled "Quad 2" and "Quad 4" in Figure 3.1. An aesthetically "nice" function which partitions these quadrants is $\sqrt{(1-R)^2 + (1-P)^2}$ which is a measure of the "distance" of the point from the ideal 100% precision, 100% recall point (the search with the smaller distance value would be "better" than the search with the larger distance value). The line drawn in Figure 3.2 was drawn by applying this distance formula to point A in the figure.

However, as mentioned earlier, recall (R) is impractical to measure, so a utility scale of the type under consideration should be based on the number of relevant retrieved documents (r). The only problem with the r/P scale is that the maximum value which r can assume varies from search to search, and cannot be determined a priori. Although this measure is likely to be different for each
user, it may well be the case that a "standardized" scale (which can be represented graphically) is a good approximation for a given class of users. One class of users, for example, may wish as many relevant documents as possible as long as precision remains at a "reasonable" level. Another example would be the class of users who desire only a few relevant documents with precision as high as possible. Thus, it may be the case that a few standardized scales can adequately represent the entire group of users for a given system.

As an example of the use of the above utility scale, a standardized r/P graph was constructed for the "average" MIC patron. This graph, presented in Figure 3.3 was developed after discussing the output requirements of MIC patrons with the MIC information specialists. The general guidelines used to construct the graph in Figure 3.3 are as follows. (1) It is generally desirable to retrieve at least a few relevant documents, even at the expense of quite low precision; this is considered to be more desirable than a very specific search which retrieves only a few documents, none of which may be relevant. (2) In general, r should be increased, even if only slightly, at the expense of precision for any precision value above 25% (approximately). (3) Since the maximum number of retrieved documents allowed by the MIC system for a retrospective search is 500, precision is more of a concern for patrons whose interests are broad enough to retrieve in excess of 500 document references.

The graph presented in Figure 3:3 should be interpreted as follows. The numbers on the far right of the graph represent ordinal
Figure 3.3. Utility Graph for Comparative Evaluation of Retrieval Outputs (Full Retrospective File Scale)
ranks of lines connecting points which are considered to represent equivalent search results. Note that the lines are drawn as a means of representation only since the space is actually discrete with only certain r/P combinations being valid; on the line of rank 1 in Figure 3.3, for example, ten discrete points are identified from points B (r=65, P=13%) to C (r=20, P=100), inclusively. Also, the above equivalences, as well as other relationships described below, apply more accurately for points which are relatively close in space. For example, more people would probably agree that point F (r=35, P=25%) is equivalent to point E (r=32, P=40%) than would agree that point F is equivalent to point C (r=20, P=100%), even though all points lie on the line with rank 1. (Fortunately when comparing search results for a given question, the points representing the search results tend to be relatively close in space). Finally, due to the 500 document maximum for retrospective searches (guideline 3 above), no points are possible in the area to the left of line 0A in Figure 3.3. If a given profile were to indeed retrieve more than 500 references, only the first 500 references would be supplied and the search results would erroneously reduce (for better or worse) to a point on the line OA.

For non-equivalent points which are reasonably close in space, any point on a line with a higher rank is considered to represent search results which are superior to those represented by a point on a line with a lower rank. Thus in Figure 3.3 point D (r=64, P=40%) which is on the line with rank 2 is considered to represent search results which are superior to those represented by point E (r=32, P=40%) on the line with rank 1.
Also when \( r \) is constant but \( P \) is increasing, the relative quality of the search results is reflected by the orthogonal distance of the points relative to the appropriate rank line(s). For example, consider the case where \( r=30 \). As the results change progressively along points G (\( r=30, P=20\% \)), H (\( r=30, P=30\% \)), I (\( r=30, P=50\% \)), J (\( r=30, P=75\% \)), and K (\( r=30, P=100\% \)), the relative quality of the search results is seen to progressively improve from points G to K in accordance with changes in the orthogonal distance of the points below and above the line of rank 1.

Since the test file to be used for interactive profile preparation with the online systems described in the next two chapters is approximately 5% of the full retrospective file, an online search which retrieves 25 document references from the test file would (ignoring sample error) retrieve 500 document references from the full retrospective file. Thus, for use with the current research, a scaled down version of the graph shown in Figure 3.3 is presented in Figure 3.4. Note that the rank lines in the two graphs correspond exactly, the only difference being that the coordinates on the \( r \) axis for Figure 3.4 are scaled down to 1/20th of those for Figure 3.3. Also note that to allow for sampling error, the lines in Figure 3.4 have been slightly extended above the line OA. This was done because a test file search which retrieves 25 or more references (no additional limit is imposed on retrieval from the test file) may retrieve less than 500 references from the full retrospective file because of sampling error. (Of course, if more than 500 references were to be actually retrieved, only the first 500 references will actually be supplied as mentioned previously).
Figure 3.4. Utility Graph for Comparative Evaluation of Retrieval Outputs (Test File Scale)
The utility graph shown in Figure 3.4 will be used in Section 7.1.7 to see how well the observed value judgments listed in Table 7.1 can be predicted from the graph. The potential value of these utility graphs is discussed in Chapter 8.
In order to demonstrate and evaluate the user aids presented in the next chapter, it was necessary to incorporate the aids into an online retrieval system. Although a few other options were considered, the system chosen was that of the Mechanized Information Center (MIC) of The Ohio State University, a batch retrieval system which was modified, for the purposes of this research, to operate in online interactive mode as well. Despite certain deficiencies, this system was chosen because it was the most readily accessible, permitted hands-on access to incorporate the desired modifications to the system, provided an adequate set of both user profiles and documents necessary to construct the user aids, and provided a steady stream of patrons who could participate in the evaluation studies. This chapter will describe the present MIC retrieval system, indicate its deficiencies in providing the desired test environment, and discuss the modifications made to remedy the perceived deficiencies.

4.1 The Mechanized Information Center and Its Retrieval Services

The Mechanized Information Center at Ohio State was established in September, 1970, by the Department of Computer and Information Science and the University Libraries. The Center began operation in February, 1971, with the goal of providing "information services that make use of machine-readable bibliographic and other data bases" (30).
Initially, MIC used an adaptation of a search system supplied by CCM Information Services, Inc, one of its database suppliers. However, the CCM software employed a character by character sequential search of the database and proved to be much too slow for the projected volume of MIC searches. MIC then adopted the software used by the Technical Information Dissemination Bureau (TIDB) of the State University of New York at Buffalo (52). This software, which uses the inverted file approach, was modified by MIC both to improve its efficiency and to fit the needs of MIC. It operates in batch mode only.

4.1.1 MIC Databases

The first database provided by MIC was a multidisciplinary file covering the physical sciences, technology, and medicine. This database was formed by merging the Pandex Current Index to Scientific and Technical Literature (obtained from CCM Information Services, Inc.), the ISI Source Tapes (obtained from the Institute for Scientific Information), the National Technical Information Service file (NTIS) (obtained through CCM), the Current Index to Conference Papers (CICP) (from CCM), and relevant records from the MARC tapes (produced by the Library of Congress). Initially, MIC provided a current awareness or selective dissemination of information (SDI) service to its patrons using the multidisciplinary database. A retrospective service using this database was begun as soon as enough information had been accumulated to make such a service worthwhile. Expansion to provide searches of other multidisciplinary databases and several discipline-based files had also
been planned. However, while some new services were introduced, the demise of several databases and database suppliers as well as the institution and growth of commercial online retrieval services has led to continued reevaluation of the original plans.

Currently, MIC provides internal search services for three databases, which are augmented by several hundred databases (53,55) accessible through the external commercial online retrieval services provided by Lockheed Information Services, Inc. (now Dialog, Inc.) (11), System Development Corporation (SDC) (46), and Bibliographic Retrieval Services (BRS) (3). This gives the MIC patron the best combination of service and value at the present time. One internal database is a multidisciplinary physical science database formed by merging the ISI and MARC tapes mentioned above (CCM Information Sciences, Inc. is now defunct). The second is a multidisciplinary social science database formed by merging ISI's Social Science Citation Index tapes with relevant records from the MARC tapes. The third is an educational database consisting of the ERIC files obtained through the National Institute of Education.

4.1.2 MIC System Software and File Structures

Originally, CCM was the main source of machine-readable information for MIC. Consequently, CCM's PANDEX format was adopted as the standard format for MIC's databases. Therefore, the tapes supplied by other vendors were converted to the PANDEX format before processing through the MIC system. The PANDEX record format consists of a variable number of varying length data elements. Each variable length record contains information concerning one bibliographic reference item.
While a detailed description of the structure of most of the MIC files is not relevant to the current research a summary of the main components of the MIC software and some of its files will now be provided accompanied by a system flow chart in Figure 4.1. The WORDGEN program (see Figure 4.1) scans each bibliographic item from an input file and produces a set of records consisting of individual title words and other information (discussed later) which are written to the Word Record File, and a record containing author, title, citation, and other information which is written to the random access Article File. Each record in the Word Record File consists of a searchable data item in character format and an article number which indicates the article in the Article File from which this item was extracted. The article numbers are sequentially assigned accession numbers which also serve as the access keys to the Article File. The searchable data items include title words (except for common words which are eliminated if they appear on a stoplist), author name, journal coden, and for some databases either thesaurus or index terms. After the Word Record File is sorted by searchable data item, (hereafter referred to as search term or descriptor), it is used by the INVERT program to produce the Inverted File, which contains all search terms, and for each, a list of the article numbers of the documents which contain the descriptor.

Another component of the MIC software processes the patrons' search requests. A patron's requirements are discussed during a conversation (commonly called an interview) between the patron and an information specialist. After the user's needs are thus
Figure 4.1. Structure of the Original MIC Multidisciplinary Current Awareness System
determined, the question is then expressed as a boolean combination of the appropriate search terms, and these terms are specified on a coding sheet in the proper format and finally punched on cards for input into the system. After the cards are checked for accuracy, they are used by the PROMAIN program (Figure 4.1) either to build a set of profiles for a retrospective search run or to update the set of profiles for a current awareness run. The MATCH program uses the Profile and Inverted Files to determine which articles are of interest to each user. The articles of interest are then retrieved from the Article File and records containing both articles and patron information are written to the Hit File sorted by patron number. For each record in the Hit File, the MICPRINT program generates a notification card containing reference information about the article, which of the user's terms caused the retrieval and, if the reference is to a journal, information concerning which of the Ohio State Libraries contains the journal. This last piece of information is obtained from the Library Location File. (There are many branch libraries at Ohio State University and journals are distributed throughout the various branches.) For the interested reader, a sample MIC notification card is shown in Figure 4.2.

4.1.3 MIC Query Language and Profile Structure

In order to provide sufficient background for a discussion of the shortcomings of the MIC system, a more detailed description of the structure of the MIC profile (search query) and its use in the retrieval process must be presented. Some examples are provided in Table 4.1 after sufficient details have been presented. Each MIC profile consists of
a series of line items (cards) containing a term group. A term group consists of one to five searchable data items with an associated group weight which ranges from .000 to .999. If all of the data items in a given group are contained in the same article, (i.e. implied AND logic between all of the terms), the group weight is assigned to the article. The weights for all groups "matching" a given article are arithmetically combined (see below) to yield a total significance value (TSV). The TSV will also lie in the range .000 to .999, and if the TSV is greater than the threshold (also in the .000 to .999 range) assigned to the entire profile, a notification card for the article is generated.

The matching function used in the MATCH program to compute the TSV was modeled after the one developed at Ames Laboratory in the mid-1960's. The Ames system was designed to permit "adaptive" profiles and was reported in the literature by Sage and his co-workers (40,41). The TIDS system used the same basic approach.
but since the threshold values and group weights were originally considered to indicate the probability that an article matching the term group was relevant to the user, the additive function used to compute a TSV by the Ames system was modified at TIDB to be a multiplicative function representing true Bayesian Probability (22). Formally, if \( n \) term groups match an article, the Total Significance Value is computed by

\[
TSV = \frac{W_1 W_2 W_3 \cdots W_n}{W_1 W_2 W_3 \cdots W_n + (1-W_1)(1-W_2) \cdots (1-W_n)}
\]  

(4.1)

As mentioned above, if the TSV is greater than the profile threshold, a notification card for the article is generated.

The motivation behind the Ames and TIDB systems was adaptability. Both the Ames and TIDB systems provided an SDI service. As a user received a notification on a porta-punch card, he returned a stub indicating whether or not the notification was relevant to his interests. The weights of the term groups generating relevant notifications were increased, while the weights of those term groups generating nonrelevant notifications were decreased. At Ames, it was discovered that the weights of terms which occurred with low frequency in the database were often decremented below the profile threshold and then could not rise above the threshold at a later time due to their infrequent occurrence. To counteract this, Ames researchers added a slow increment function to bring such terms back into consideration. The use of this technique at TIDB led to a different problem. In that environment, the slow increment function
tended to push high frequency terms over the threshold and produce a large number of false drops (1).

Based on the experience with the software at TIDB, although the format of the profile was not changed by the MIC staff, the method of use was altered significantly. A "standard" set of weights was developed which could be assigned to term groups to produce the desired boolean combination of the groups. This technique, based on a profile threshold of .750, is described in both Cobes (6) and MIC (30). Eventually, it became the practice to code profiles in what might be referred to as sum-of-products form. Currently, the profile and all term groups are assigned a weight of .750, so that any article which matches one or more term groups is retrieved. Thus, AND logic is implied between the terms within any term group, while OR logic is in effect between the different term groups due to the weighting scheme. A simple example of the use of this scheme is illustrated for example 1 in Table 4.1. While the profiles produced by this method often appear awkward, especially if they are more naturally coded in product-of-sums form as shown for example 2 in Table 4.1, the use of a shorthand notation on the coding sheet coupled with a knowledge of this shorthand by the keypuncher have produced a fairly efficient use of this system at MIC. (An alternate scheme which is easier to use for certain types of profiles has been developed by Petrarca (36) and is presented in Appendix D). A type of NOT logic is possible by assigning a term group a weight of .000. However, due to the nature of the matching function, this
Table 4.1
A Few Examples of Coding a Profile in MIC Format

<table>
<thead>
<tr>
<th>Boolean Request</th>
<th>MIC Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) INFORMATION AND (STORAGE OR RETRIEVAL)</td>
<td>Weight</td>
</tr>
<tr>
<td></td>
<td>.750</td>
</tr>
<tr>
<td></td>
<td>.750</td>
</tr>
<tr>
<td>(2) (DATA OR INFORMATION OR FILE) AND (STRUCTURE OR ORGANIZATION)</td>
<td>.750</td>
</tr>
<tr>
<td></td>
<td>.750</td>
</tr>
<tr>
<td></td>
<td>.750</td>
</tr>
<tr>
<td></td>
<td>.750</td>
</tr>
<tr>
<td></td>
<td>.750</td>
</tr>
<tr>
<td></td>
<td>.750</td>
</tr>
<tr>
<td>(3) (CARS NOT CHRYSLER) OR (LIGHT AND TRUCKS)</td>
<td>.750</td>
</tr>
<tr>
<td></td>
<td>.000</td>
</tr>
<tr>
<td></td>
<td>.750</td>
</tr>
<tr>
<td>(4) (CARS NOT CHRYSLER) OR (LIGHT AND TRUCKS)</td>
<td>.750</td>
</tr>
<tr>
<td></td>
<td>.000</td>
</tr>
<tr>
<td></td>
<td>.750</td>
</tr>
</tbody>
</table>
type of NOT applies to the entire profile and must be used with care. This is illustrated in example 3 of Table 4.1 for a request concerning all cars except those made by Chrysler, and all light trucks including those made by Chrysler. The MIC profile illustrated for example 3, however, does not meet this request since the .000 weight assigned to Chrysler will apply to light trucks also, and no information about light trucks made by Chrysler will be retrieved. A better MIC profile for this question is shown in example 4 of Table 4.1. In this case, information concerning Chrysler light trucks will be found, unless the document also discusses Chrysler cars. Although even this MIC profile does not exactly match the boolean request, it is much closer than the first profile. An exact match is possible using the procedures developed by Petrarca given in Appendix D. (See Figure D.1, example 1.)

4.2 The Standard Online System

As originally envisioned, the aids were to be a set of procedures designed to automatically enhance an MIC profile based on information which could be extracted from the set of current awareness profiles and the documents being searched. However, with online retrieval systems moving out of the experimental and into the operational stage, it was felt that developing enhancements for a batch system and then testing to compare a standard batch system with the enhanced batch system might well prove to be a largely wasted research effort. It can easily be argued that even an enhanced batch system is inferior to a standard online interactive
system simply due to the inherent differences between batch retrieval and online retrieval.

It was therefore decided to develop aids useful in an online environment and to compare a state of the art online system with an online system containing the user aids. Since the MIC system was to be used for the experiment, a necessary first step was to design and implement software capable of accessing the MIC files in an online, interactive environment. Further, the online system thus implemented should resemble a state of the art operational retrieval system as closely as possible while maintaining compatibility with the batch system. However, considering that this task, although requiring a major and very time consuming effort, provided little more than a baseline retrieval system, certain minor deviations from the state of the art were made. These minor deviations are discussed in the next section (4.3) after the online system (from now on referred to as the Standard system) is described, and their impact on the results of the research is discussed in Section 7.1.6.

The first version of the online system was implemented by rewriting the MATCH program (refer to Figure 4.1) to accept a profile input at the terminal, search the database, and finally display the set of results at the terminal. Again referring to Figure 4.1, this version of the MATCH program performed all of the functions performed by MATCH, PROMAIN, and MICPRINT in the batch system. The profile was input in a modified MIC format. To provide compatibility with standard MIC profiling practice a weight of .750 was assumed for each term group, and a threshold of .750
was assumed for the profile. On one line the user would enter the number of terms in the term group, and on the following lines he would enter the individual terms, one per line. This process was repeated until the entire profile had been entered. The titles of the retrieved articles were then displayed in a compact format after the files had been searched. (While the batch system display format is nice for notification cards, it is both time consuming and awkward when used for online display.)

Noting the unacceptability of coding an entire profile before obtaining any results (the first version might be viewed as a batch system with very fast turn around time), a second version of the online system was written. This version performed a search of the file as each term group was entered and immediately displayed a posting count of retrieved articles for the term group. The input was improved to accept the entire term group on a single line with spaces separating the individual terms. The count indicating the number of terms in the group was no longer required. The printing was no longer automatic, but was performed when the user entered a somewhat elementary print command. Five titles were displayed for the chosen retrieval set (if that many articles were in the set), and the user had the continuing option of printing the next five or returning to the search pattern until the list of titles was exhausted.

While the second version of the online system represented a distinct improvement over the first, several problems still existed,
and these were corrected in the third and final version of the Standard system used for this study. First, the profiling language was modified. Although the MIC profiling language works fairly well in the batch environment, the awkwardness and profile length which are inherent in the sum-of-products form are intolerable at the terminal, especially for users with poor to fair ability at the terminal keyboard. The new language can best be described as a standard boolean logic language. The simplest form of query to search the file is to enter simply a single term. The system responds with the number of postings and an echo of the term. The result set of retrieved article numbers for each search line is saved as the search progresses. Terms may be combined on a single search line by using infix boolean operators (AND, OR, NOT which may be abbreviated A, O, N). The implicit precedence is AND, OR, NOT, and each operator is applied left to right on the search line (i.e. all of the AND's are evaluated first, with the leftmost AND evaluated before the others). In addition, parentheses may be used to change the default order of evaluation, with expressions enclosed in parentheses being evaluated first. Since the result set for each line is saved, line numbers may be used to represent lines to be combined with other lines, newly entered terms, or a combination of the two by using the boolean operators. The above features are illustrated in lines 1-4 of the sample terminal session illustrated in Figure 4.3.
Figure 4.3. A Sample Terminal Session - Standard System
The PRINT command was also enhanced to make it simpler and more useful. This version is modeled rather loosely after the BRS "..PRINT" command (3) and has the following syntax:

```
PRINT   line#   format   /items
```

PRINT may be abbreviated P, and line#, if omitted, defaults to the most recent line entered. The format allows the user to select certain combinations of author, title, and citation information; the default is title only. The items field is used to select the documents from the set for which the information is to be displayed. The default for this field is to display the information for the first five documents in the set, or for all documents if there are fewer than five. The use of the print command is also illustrated in the sample terminal session presented in Figure 4.4. A more detailed description of the above features, additional features, and an additional terminal session are presented in the MIC-Online User's Manual reproduced as Appendix A.

4.3 The Standard System Versus the State of the Art

As mentioned in the last section, there are a few features contained in a state of the art retrieval system which are not included in the Standard system. First, there is no command which allows the user to browse through the inverted file index (e.g., ORBIT's "NEIGHBOR" command (46)). A second feature not available is the display of all terms retrieved by a truncated term. Finally, when a string of terms is entered on the same search line, the postings for the individual terms are not displayed. The user
desiring the postings for the individual terms must enter one term per line and then combine the terms by using the line numbers. The impact, if any, of these omissions on the current research is discussed in Section 7.1.6.

Now that the baseline environment has been described, the next chapter will describe the search system containing the user aids, and Chapter 6 will then describe the procedures for evaluation of the search aids.
CHAPTER V. THE AIDS SYSTEM AND THE INDIVIDUAL AIDS

As discussed in Chapter 1, the main thrust of this research is to develop and evaluate online user search aids. Four such aids designed to assist the user in coding a better search profile have been added to the Standard system to produce what will be referred to as the Aids system. This chapter will describe the Aids system in six sections. The first section will describe the four aids from a conceptual point of view and provide the theory and motivation behind each. The second through fifth sections will provide more technical descriptions of each aid and discuss the processes by which each aid was constructed. The final section will present an overall view of the Aids system as seen by the user.

5.1 The Concepts Behind the Aids

In order to develop an effective search profile for a document retrieval system there are two basic requirements: the user must have a minimum amount of knowledge concerning how the system works and he must have expertise in the subject area he is searching in order to know how best to phrase the query to the system. However, it is a difficult task, even for a knowledgeable expert, to think of all appropriate terms which may be used in the document collection to describe the subject of his search. Three of the profiling aids are designed to help the user isolate those terms which are appropriate for his search.
The first such aid, to be called the Profile-Profile Association Aid, will enable a user to locate and display any existing profile which is similar to his own so that he can make use of the entire profile or any pertinent part of the profile. As discussed in Chapter 1, in any large retrieval system there are new questions which are identical or at least similar to previously asked questions, but the knowledge and experience gained by other users who have asked similar questions are currently available to the user only if an information specialist can both remember that the question has been asked previously, and then locate the profile which corresponds to the question. This first aid will make such valuable information available to any user who can at least code a simple initial profile.

Although the above profiling aid can be very valuable, it may well be the case that no existing profile, taken as a whole, is similar to the new profile; therefore, the second aid will allow the user to make use of the intellectual effort invested in the set of existing profiles in a more general manner. This profiling aid, to be called the Term-Term (Profile) Association Aid, will help the user to expand the search terms in his profile by displaying any earlier profile terms (not necessarily from a single profile) related to the user's initial terms as determined by term-term associations exhibited by the various terms in the existing set of profiles.

However, in some cases a new profile will not even remotely resemble any existing profile, so neither of the above aids will be of any value to the user coding such a profile. The third aid,
the Term-Term (Document) Association Aid, will assist this type of user in the expansion of his initial subject terms by displaying new terms which are related to the initial terms as determined by term-term associations computed from a sample file of documents representative of the entire document collection.

Although the above three aids may suggest terms to the user to help him narrow or focus his search, in most cases these aids will suggest synonyms and other related terms which will assist the user in broadening his search to assure adequate coverage of each concept. However, the fourth aid, called the Relevance Feedback Aid, is designed to help the user focus his search. This aid, as its name suggests, uses a relevance feedback approach whereby the user may indicate which completed section of the search he wishes to examine, and the system will retrieve the indicated documents. After the user indicates which of the documents are relevant and which are not relevant to his interests, the system will examine the terms within each document and try to detect any patterns which may be important to the user. For example, a term which has not been included in the search strategy, but which appears in many of the relevant documents, and in few or none of the nonrelevant documents, is a good candidate to be used as a term to be combined with the others in the strategy by using AND logic. Similarly, a term which has not been included in the strategy, but which appears primarily or exclusively in nonrelevant documents is a good candidate to include by using NOT logic.
In summary, the four aids are designed to work in harmony. The first three are designed to insure adequate breadth, and the fourth is designed to assist in providing proper focus. The next section will discuss each aid in more detail and will also provide information concerning the actual implementation of each aid.

5.2 The Profile-Profile Association Aid

As mentioned earlier, the first aid is designed to discover already existing profiles which are similar to the profile being coded, thereby taking advantage of the intellectual effort already expended by other users. In general, this aid uses the association measure techniques discussed in Chapter 2 to accomplish the objective.

At first, the Tanimoto measure discussed in Section 2.5 was used to measure the association between the new and existing profiles. (This particular measure was chosen because of its successful use in related earlier research.) As applied to profile-profile association, this measure takes the following form:

\[
S_T(Q_i, Q_j) = \frac{\sum_{k \in l} v_{ik} v_{jk}}{\sum_{k \in l} v_{ik} + \sum_{k \in l} v_{jk} - \sum_{k \in l} v_{ik} v_{jk}}
\] (5.1)

\(S_T(Q_i, Q_j)\) is the measure of association or similarity between profile \(Q_i\) and profile \(Q_j\). The value of \(S_T\) ranges from 0.0 to 1.0, and the closer the value of \(S_T\) is to 1.0, the more similar are the profiles. The total number of unique terms contained in the entire set of profiles being considered is represented by \(t\). The value of \(v_{ik}\) is 1 if term \(k\) is in profile \(Q_i\), and the value of \(v_{ik}\) is 0 if
term $k$ is not in profile $Qi$. Since $v_{ik} v_{jk}$ is 1 only if term $k$ is contained in both profile $Qi$ and profile $Qj$, $S_T (Q_i, Q_j)$ is a measure of the number of unique terms which occur in both profiles divided by the number of unique terms which occur in either or both of the profiles.

A simple example which further illustrates this concept is presented in Figure 5.1. If we assume, for the sake of illustration, that profiles $Q_1$, $Q_2$, and $Q_3$ in the figure define the entire set, a unique number can be (arbitrarily) assigned to each unique term. The profiles can then be represented by vector notation as shown in the figure and discussed in Section 2.5. The computation of the association measure between each pair of profiles is indicated at the bottom of the figure. The reader should note the similarity between this computation and that presented in Figure 2.1(b) for document-profile association.

A problem which arises when using the Tanimoto measure to determine the association between a new profile and an existing one is that the value of the measure is dependent on the lengths of the two profiles. For example, if a user coding a new profile (profile $Q_4$) enters INFORMATION AND RETRIEVAL to begin his search, $S_T (Q_1, Q_4) = .67$ while $S_T (Q_3, Q_4) = .33$, where profiles $Q_1$ and $Q_3$ are defined as in Figure 5.1. While this dependence on length is desirable in many applications and works well for comparing two complete profiles, it is not an advantage in the context of a profiling aid. In coding a profile, it is desirable to examine a
Profile Q1: INFORMATION AND (STORAGE OR RETRIEVAL)
Profile Q2: INFORMATION AND SYSTEMS AND (MANAGEMENT OR ADMINISTRATION)
Profile Q3: (INFORMATION OR DOCUMENT) AND (STORAGE OR RETRIEVAL) AND (SYSTEMS OR SYSTEM)

Assign the following numbers to the terms:

<table>
<thead>
<tr>
<th>Term</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1) ADMINISTRATION</td>
<td>T5) RETRIEVAL</td>
</tr>
<tr>
<td>T2) DOCUMENT</td>
<td>T6) STORAGE</td>
</tr>
<tr>
<td>T3) INFORMATION</td>
<td>T7) SYSTEM</td>
</tr>
<tr>
<td>T4) MANAGEMENT</td>
<td>T8) SYSTEMS</td>
</tr>
</tbody>
</table>

Q1 = (0,0,1,0,1,1,0,0)
Q2 = (1,0,1,1,0,0,1)
Q3 = (0,1,1,0,1,1,1,1)

Then \[ S_T (Q1, Q2) = \frac{1}{3+4-1} = \frac{1}{6} = .1666 \]
\[ S_T (Q1, Q3) = \frac{3}{3+6-3} = \frac{3}{6} = .5000 \]
\[ S_T (Q2, Q3) = \frac{2}{4+6-2} = \frac{2}{8} = .2500 \]

Figure 5.1. Computing Profile-Profile Associations Using the Tanimoto Measure (Equation 5.1)
similar profile (if one exists) as quickly as possible. If an existing similar profile (such as profile Q3) has even a moderate number of terms, the association value will remain rather low until much of the new profile has been entered, thereby diminishing the effectiveness of this aid.

To counteract this problem, a different measure was tested: the asymmetric measure described in Salton (42). This measure can be defined by

\[
S_A(Q_i, Q_j) = \frac{\sum_{k=1}^{t} v_{ik} v_{jk}}{\sum_{k=1}^{t} v_{ik}}
\]  

(5.2)

where \(v_{ik}, v_{jk}\), and \(t\) are defined as above for the Tanimoto measure. \(S_A(Q_i, Q_j)\) again ranges in value from 0.0 to 1.0, and is actually a measure of the fraction of profile \(Q_i\) which is a subset of profile \(Q_j\) (hence this measure will be called the Asymmetric Subset measure). This measure is asymmetric since, in general, \(S_A(Q_i, Q_j) \neq S_A(Q_j, Q_i)\). (The Tanimoto measure is a symmetric measure since \(S_T(Q_i, Q_j) = S_T(Q_j, Q_i)\) in all cases.) Sample computations for this measure are presented in Figure 5.2.

Therefore, if we compute \(S_A(Q_i, Q_j)\) where \(Q_i\) is the new profile and \(Q_j\) is an existing profile, we have a value representing the degree to which the terms in the new profile are contained in the existing profile. Since the length of the existing profile does not influence \(S_A\), this measure can quickly isolate existing profiles which are similar to the new one. For example, comparison of the new profile Q4 (INFORMATION AND RETRIEVAL) with Q1, Q2, and
Q1 = (0,0,1,0,1,1,0,0)  
Q2 = (1,0,1,1,0,0,1)  
Q3 = (0,1,1,0,1,1,1,1)  

\[ S_A(Q1, Q2) = \frac{1}{3} = .333 \]  
\[ S_A(Q2, Q1) = \frac{1}{4} = .250 \]  
\[ S_A(Q1, Q3) = \frac{3}{3} = 1.000 \]  
\[ S_A(Q3, Q1) = \frac{3}{6} = .500 \]  
\[ S_A(Q2, Q3) = \frac{2}{4} = .500 \]  
\[ S_A(Q3, Q2) = \frac{2}{6} = .333 \]  

**Figure 5.2.** Computing Profile-Profile Associations  
Using the Asymmetric Subset Measure (Equation 5.2)  
for the Profiles Defined in Figure 5.1

Q3 from Figure 5.1 gives  
\[ S_A(Q4, Q1) = 1.0, \quad S_A(Q4, Q2) = .50, \quad S_A(Q4, Q3) = 1.0. \]  
This suggests that Q4 exhibits strong correspondence to both Q1 and Q3 which were not immediately evident through use of the Tanimoto measure above. For this reason, the Asymmetric Subset measure has been more useful than the Tanimoto measure for the purpose of this aid.

Operationally, this measure has been implemented in the Aids system in the following manner. A set of 342 profiles (both current awareness and retrospective) was isolated as the test set to be used for the profile-profile aid. Each profile in the test set was assigned a number (1-342), which is used to reference the profile inside the Aids program. From this set of profiles, which contains 5,055 unique terms, an inverted file has been constructed with each profile term as an entry point. Following each term is a list of the numbers representing the profiles which
contain the term. A 342 word array is used to maintain a count of the number of terms which occur in both the new profile and an existing profile (the $\sum v_{ik} v_{jk}$ term in Equation 5.2). A count of the number of terms which have been entered for the new profile is also kept (the $\sum v_{ik}$ term in Equation 5.2). Thus, moving through the array and performing one division for each element quickly yields a measure of association between the new profile and each of the existing profiles. The conditions under which a profile is displayed are discussed below while the actual file structures used in the Aids program and the algorithms used to trigger the various aids are presented in Appendix B. Also, a system flow chart with an accompanying description of how the files for the aids were constructed is shown in Appendix C.

For the purposes of the present research, the conditions under which this aid is triggered are as follows. An existing profile is considered for display only if the value of $S_A$ is .70 or greater. This cutoff value, which was determined by preliminary testing, might be slightly too high for many applications since related profiles sometimes exhibit association measures below .70; but because the profile display can be slightly lengthy and because relatively slow terminals were being used, it was desired to display only those profiles with a very good chance of being helpful. Also, since the $S_A$ value after the first new profile term has been entered is 1.0 for each existing profile containing the entered term, the $S_A$ value is meaningless until a minimum of
two new profile terms have been entered. It is even questionable whether two terms provide enough information to determine whether two profiles are sufficiently similar. During preliminary tests, it was found that reasonable discrimination began with profiles containing three terms. Therefore, the testing for similar profiles begins after three terms have been entered by the user. Illustration of how this aid actually works in the context of an actual search is deferred until Section 5.6, after the other aids have been discussed.

Before ending the discussion of this aid, a problem encountered in profiles using truncated terms should be mentioned. A truncated term is usually used to represent several terms with a common root. While it is reasonable to match BROTHER and BROTHER* (the asterisk represents the truncation symbol) and count this as a co-occurrence, other questions immediately arise. If this is counted as one co-occurrence, and BROTHER is counted as a single term for the first profile, should BROTHER* be counted as one term or several for the second profile. (It actually represents BROTHER, BROTHERS, BROTHERLY, BROTHERLINESS, BROTHERHOOD, and possibly several other terms.) The value of the association between two profiles can thus vary greatly depending on the measure used and the interpretation assigned to the truncated terms. Although a truncated term is counted as a single term for the current implementation, better solutions undoubtedly exist. This problem is identified as one which warrants additional study in Chapter 8.
5.3 The Term-Term (Profile) Association Aid

The second aid is also designed to take advantage of the intellectual effort already expended by the users, and this aid is more generally applicable, but possibly less useful than the first aid. For this aid, an association value has been computed between each unique pair of terms contained in the test set of 342 profiles. For any pair of terms which exhibit a high similarity as measured by the association between terms, it is quite likely that a user entering one of the terms would wish to use the other as well.

For this aid, the Tanimoto measure proved to work well. In this context, the measure takes the following form:

\[ S_A (T_i, T_j) = \frac{\sum_{k=1}^{q} w_{ik} w_{jk}}{\sum_{k=1}^{q} w_{ik} + \sum_{k=1}^{q} w_{jk} - \sum_{k=1}^{q} w_{ik} w_{jk}} \] (5.3)

In Equation 5.3, \( S_A (T_i, T_j) \) is the Tanimoto measure of association between terms \( T_i \) and \( T_j \), while \( \sum_{k=1}^{q} w_{ik} w_{jk} \) represents the number of profiles which contain both term \( T_i \) and term \( T_j \), and \( \sum_{k=1}^{q} w_{ik} \) represents the number of profiles which contain term \( T_i \). Finally, \( q \) is the number of profiles in the entire set of profiles.

Note that this is identical to computing term-term associations as discussed in Section 2.5, only in this case profiles are used instead of documents. A few examples of term-term (profile) association values computed from the set of 342 profiles are displayed in Table 5.1.
Table 5.1
Example of Term-Term (Profile) Association Values

<table>
<thead>
<tr>
<th>TERM #1&lt;sup&gt;a&lt;/sup&gt;</th>
<th>TERM #2&lt;sup&gt;ab&lt;/sup&gt;</th>
<th>ASSOCIATION VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADOLESCEN*</td>
<td>ALCOHOL*</td>
<td>.2857</td>
</tr>
<tr>
<td></td>
<td>CHILD*</td>
<td>.0769</td>
</tr>
<tr>
<td></td>
<td>DELINQUEN*</td>
<td>.2500</td>
</tr>
<tr>
<td></td>
<td>DRUG*</td>
<td>.3333</td>
</tr>
<tr>
<td></td>
<td>JUVENILE*</td>
<td>.3999</td>
</tr>
<tr>
<td></td>
<td>MARIJUANA</td>
<td>.3333</td>
</tr>
<tr>
<td></td>
<td>MIDDLE</td>
<td>.1666</td>
</tr>
<tr>
<td></td>
<td>SEX*</td>
<td>.1333</td>
</tr>
<tr>
<td></td>
<td>SUICID*</td>
<td>.2857</td>
</tr>
<tr>
<td></td>
<td>TEEN*</td>
<td>.6666</td>
</tr>
<tr>
<td></td>
<td>YOUNG</td>
<td>.3333</td>
</tr>
<tr>
<td></td>
<td>YOUTH*</td>
<td>.5000</td>
</tr>
<tr>
<td>BROTHER*</td>
<td>SIBLING*</td>
<td>.7500</td>
</tr>
<tr>
<td></td>
<td>SISTER*</td>
<td>.7500</td>
</tr>
<tr>
<td>CAMPAIGN</td>
<td>MEDIA</td>
<td>.1052</td>
</tr>
<tr>
<td></td>
<td>POLITIC*</td>
<td>.1999</td>
</tr>
<tr>
<td></td>
<td>TELEVISION</td>
<td>.1110</td>
</tr>
<tr>
<td></td>
<td>VOTE*</td>
<td>.6666</td>
</tr>
<tr>
<td></td>
<td>VOTING</td>
<td>.5000</td>
</tr>
</tbody>
</table>

<sup>a</sup>The problem associated with truncated terms discussed in Section 5.2 for the Profile-Profile Association Aid effects the Term-Term (Profile) Association Aid in a similar manner. (The asterisk after a term represents the truncation symbol.)

<sup>b</sup>The ordering in this table is alphabetical; as used in the aid, the ordering is from highest association value to lowest.
As mentioned in Section 2.6, one problem with term-term association based on document titles is that synonymous words do not tend to appear in the same title, and the terms with high association values tend to be contiguous words (words which often occur in the same phrase - e.g. United and States). With term-term associations based on profiles, several types of relationships should exhibit relatively high association. One will be the contiguous relationship mentioned above; this will be the result of two terms often connected with AND logic (e.g. United AND States). However, synonyms should also exhibit high association since a good profile for searching a free text system should include all terms which can be used to express a concept of interest (i.e. all synonyms). Several pairs in Table 5.1 illustrate this, one example being the ADOLESCEN*/TEEN* pair. In addition, generic terms and terms at the same level of a hierarchy should exhibit a high association value in many cases. For example, a search for FARM ANIMALS in a free text system would probably include the more specific terms COW, PIG, and CHICKEN as well as terms for other specific farm animals. While these terms are obviously not synonyms, a user coding a new profile who enters the term PIG, may or may not be interested in the terms COW and CHICKEN, but he should be interested in the generic term FARM ANIMALS (even if the user is specifically interested in PIGS, since some information on PIGS may be included in documents dealing with FARM ANIMALS in general). The BROTHER*/SISTER*/SIBLING* group presented in Table 5.1 is a good illustration of this type of relationship.
Operationally, the term-term association values were first computed for all pairs of terms in the set of profiles. If a term occurred in only one profile, it was eliminated from consideration, since many terms which occur in only one profile exhibit high associations to other low frequency terms due to the nature of the association measure. While two such terms are sometimes related, more often they are not, and the low frequency tends to place such terms at the top of the association list causing other, more accurate associations to drop further down the list. The results of this phenomenon, caused by low frequency and a form of sampling error, are considered to be undesirable. The easiest way to solve the problem is to simply eliminate, before computing the associations, those terms which occur in only one profile. This removes from the association list the often inaccurate associations found between low frequency terms, and considerably reduces the size of the association file.

After this first step, all pairs with a low association value (below .250) were also eliminated. An examination of the associated term pairs had shown that, in general, if the association between the pairs were below .250, then the pairs were not related. By removing such term pairs, the user would be not be burdened with examining useless pairs, and the size of the association file was further reduced. The remaining pairs were saved for use by the aid. A form of inverted file has been constructed for this aid also. The entry point is again the term, and following each term is a list
of all associated terms ordered from highest association value to lowest. Thus, unlike the profile-profile measure which must be computed by the Aids program, the computation for this aid has been performed previously, and the related terms can be retrieved by simple table lookup. Programming considerations are again presented in Appendices B and C.

5.4 The Term-Term (Document) Association Aid

Although this type of association has been studied since the mid-1960's, it has been included in this research for several reasons. First, for much of the work which has been done in this area, the size of the document set used to form the associations has been relatively small (500-10,000 documents depending on the study). As mentioned in Section 2.6, one reason for this is that the matrix technique used to generate the associations is neither efficient nor cost effective. By using a type of sparse matrix and modifying the processing algorithm for the current research, almost 25,000 documents have been processed in an efficient manner to form the term-term associations. This method of processing is described in Appendix C. (Parts of this method were first suggested by Conchita Beaton (2). I later found a similar method proposed in the literature in a brief communication by Cagan (5).)

The second reason for including this aid is based on the premise that for many questions the first two aids presented will be of no help because the set of profiles simply does not contain a profile which deals with the new topic being searched (although this case should
arise less frequently if the size of the set of stored profiles is increased). In this case, the term-term (document) aid will be the only aid of potential value to the user.

Finally, it will be informative to compare the performance of the two different term-term aids, both in the scope of coverage and in quality of associations produced.

To construct this aid a learning file of approximately 5% of the MIC Social Science retrospective database was used. The 24,878 documents in this file were uniformly selected from a set of documents covering a six month period, thereby insuring that quarterly journals would be adequately represented in the sample.

For this aid the Tanimoto measure was also used, and it was applied as described in Section 2.5. The operational aspects are similar to those described in the last section. Terms occurring in only one document were eliminated. However, instead of a .250 cutoff, a value of .01 was used to eliminate low association values. This was necessary since the large number of documents has tended to push these association values into a much lower range than that found with the associations based on profiles. The cutoff value of .01 was determined by prior inspection of the association lists. Although many associations of poor quality exist at low association values, some reasonable associations were found at this level. Since the pairs with higher values are displayed first it was decided to include the lower values simply to determine their effect even though they would not be displayed in many cases. The file structures
and programming considerations for this aid are similar to those discussed in the last section (see details in Appendices B and C). Table 5.2 presents some term-term association values computed for terms from the document set.

5.5 The Relevance Feedback Aid

After the other aids are used to broaden the search, this aid is designed to help the user focus his search. For a selected search line, the system will display the titles of documents retrieved by the search line. The user then enters "R" or "Y" if the document is relevant to his search, "N" if it is not relevant, and "D" if he doesn't know (for example, a foreign language title).

The system then tallies the number of relevant and nonrelevant documents in which each term in the set of retrieved documents is contained. The titles which were rated "D" are not considered. Separate lists of candidate terms and appropriate Boolean logic for combining with the search line to narrow down the search are then displayed.

For example, assume on search line 4 the user enters the following:

LINE 4: program and planning

If he obtains three hits, and LINE 4 is then to be evaluated, the relevance feedback aid would operate as shown in Figure 5.3. In this simple example, the AND terms should indicate to the user that the initial search can be focused by adding one or more of the terms with AND logic. If this is undesirable, the NOT group indicates terms which may be added using NOT logic to remove at least some of
Table 5.2
An Example of Term-Term (Document) Association Values

<table>
<thead>
<tr>
<th>TERM #1</th>
<th>TERM #2&lt;sup&gt;a&lt;/sup&gt;</th>
<th>ASSOCIATION VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADOLESCENT</td>
<td>ADOLESCENCE</td>
<td>.0545</td>
</tr>
<tr>
<td></td>
<td>CHILD</td>
<td>.0499</td>
</tr>
<tr>
<td></td>
<td>CHILDREN</td>
<td>.0091</td>
</tr>
<tr>
<td></td>
<td>GIRLS</td>
<td>.0769</td>
</tr>
<tr>
<td></td>
<td>JUVENILE</td>
<td>.0092</td>
</tr>
<tr>
<td></td>
<td>PARENT</td>
<td>.0053</td>
</tr>
<tr>
<td></td>
<td>SEX</td>
<td>.0195</td>
</tr>
<tr>
<td></td>
<td>SOCIALIZATION</td>
<td>.0405</td>
</tr>
<tr>
<td></td>
<td>YOUTH</td>
<td>.0289</td>
</tr>
<tr>
<td>BROTHERS</td>
<td>RHYME</td>
<td>.1110</td>
</tr>
<tr>
<td></td>
<td>SISTERS</td>
<td>.3529</td>
</tr>
<tr>
<td></td>
<td>STORIES</td>
<td>.0208</td>
</tr>
<tr>
<td>CAMPAIGN</td>
<td>ELECTIONS</td>
<td>.0499</td>
</tr>
<tr>
<td></td>
<td>FUNDS</td>
<td>.1562</td>
</tr>
<tr>
<td></td>
<td>POLITICAL</td>
<td>.0064</td>
</tr>
</tbody>
</table>

<sup>a</sup>The ordering in this table is alphabetical; as used in the aid, the ordering is from highest association value to lowest.
LINE 4: program and planning

<table>
<thead>
<tr>
<th>TERM</th>
<th>#REL</th>
<th>#NOT REL.</th>
<th>PRECISION(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1) COMMUNITY</td>
<td>1</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>2) MENTAL</td>
<td>1</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>3) HEALTH</td>
<td>1</td>
<td>0</td>
<td>100</td>
</tr>
</tbody>
</table>

POSSIBLE "NOT" TERMS FOR LINE 4

<table>
<thead>
<tr>
<th>TERM</th>
<th>#REL. LOST</th>
<th>#NOT REL. ELIM.</th>
<th>PRECISION(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1) CAREER</td>
<td>0</td>
<td>2</td>
<td>33 100</td>
</tr>
<tr>
<td>2) DEVELOPMENT</td>
<td>0</td>
<td>1</td>
<td>33 50</td>
</tr>
<tr>
<td>3) UNDERGRADUATES</td>
<td>0</td>
<td>1</td>
<td>33 50</td>
</tr>
<tr>
<td>4) IMPLEMENTING</td>
<td>0</td>
<td>1</td>
<td>33 50</td>
</tr>
<tr>
<td>5) LIFE</td>
<td>0</td>
<td>1</td>
<td>33 50</td>
</tr>
</tbody>
</table>

Figure 5.3. An Example of the Relevance Feedback Aid
the unwanted documents. The reader should note that, in general, at least seven to ten documents are required for this aid to be of much value to the user.

5.6 The Aids System - User's View

Now that the concepts behind the aids as well as the aids themselves have been discussed, the overall Aids system will be presented. The system is presented as it has been structured to gather the data required for this research. Possible changes to make the aids more suitable for an operational environment are indicated at the end of this section.

Each search line is entered and the postings are displayed in the same manner as they are in the Standard system. At this point, each term on the line is checked against the two association term lists and if the lists are not empty, the associated terms are displayed in the format shown in Figure 5.4. In this figure, it can be seen that after the user enters the term "death" on line 4, first the posting line is displayed, and under the posting line the input term is indicated (necessary for lines containing more than one term) followed by the two association lists with the terms from the profiles being displayed in the first list. A maximum of five terms are displayed on each list, and if any terms remain, the user is given the option of displaying more. If the user enters any term more than once during the session, the association lists are displayed only the first time the term is used. In Figure 5.4, there are only two terms in the first list, but there are more than five
in the second. Thus, the two terms in the first list are displayed, as are the first five in the second list. At this point, the user may type "M" in response to the system prompt and the system will display the next five terms in the second list (if that many still remain) and again issue the prompt. When the user feels that he has examined enough terms from the lists, he simply responds to the prompt by entering a carriage return. The system will then begin to display the associations for the next term entered on the current line, or if all terms on the current line have been considered the system will issue the prompt to begin the next search line (e.g. LINE 5:).

Figure 5.4. A Sample of the Term-Term Aids

After at least three terms have been entered by the user, a similar profile, if one exists, is displayed for the user. Suppose, for example, that the user is interested in the recent wars in the Mid-East, and on his first search line enters

LINE 1: WAR AND (ISRAEL* OR EGYPT OR ARAB)
Since at least three terms have been entered, a check for similar profiles will be made. If a similar profile is found, it will be displayed as shown in Figure 5.5. In this figure, a similar profile is displayed in a modified MIC format. While this is not actually desirable, the existing profiles are all in MIC format and everyone using the system for this study was familiar with the format. As described in the last chapter, OR logic is implied between the various lines, while AND logic is implied between terms on the same line. After seven lines have been displayed, the user is given the option of continuing to examine the profile or returning to the search pattern. In Figure 5.5, it is assumed that the term "WAR" is contained in the part of the profile which has not been displayed. Thus, with three terms in common, the $S_A$ measure between the two profiles would be .750 (since four terms have been entered for the new profile $S_A = \frac{3}{4} = .750$). Since this is above the .70 cutoff, the profile in the example would be displayed for the user.

**Figure 5.5. A Sample of the Profile-Profile Aid**

LINE 1: war and (israel* or egypt or arab)

6 (126) WAR AND (ISRAEL* OR EGYPT OR ARAB)

THE FOLLOWING PROFILE IS SIMILAR TO YOURS:

ARAB
CONFLICT
ISRAEL*
JEW
JEWISH
MID EAST
PALESTINE

TYPE "M" TO SEE MORE ELSE HIT RETURN:
When the user has indicated completion of the profile by enter­
ing a carriage return in response to a line prompt, he receives the
message

ENTER LINE NUMBERS TO BE INCLUDED:

to which he should reply by listing the numbers of the lines contain­
ing final strategies (as opposed to intermediate results). The
system begins relevance feedback processing by first examining the
line entered above with the largest number of postings. The proce­
dure described in Section 5.5 is then begun. The display is
identical to that shown in Figure 5.3 and will not be repeated here.
After each line completes the feedback processing, the user has the
option of examining with relevance feedback the line with the
largest number of postings remaining, going back to the search
process to modify his strategy, or concluding the search.

In an operational environment, these aids should probably be
invoked by user command instead of by the automatic algorithms
described above. One command should instruct the system to "display
the associated terms for the following term." Another should
instruct the system to check for similar profiles, and a third should
"begin the relevance feedback procedure" for the indicated line
number. This flexibility has not been provided in the experimental
system in order to gather as much data as possible to use in evaluat­
ing the aids. The evaluation procedure is discussed in the next
chapter.
CHAPTER VI. EXPERIMENTAL DESIGN OF EVALUATION PROCEDURES

An important goal of this research was to develop procedures to evaluate the four online interactive profiling aids discussed in the last chapter. Section 6.1 below discusses some important design considerations for such an evaluative procedure; Section 6.2 presents two design alternatives for composite evaluation of the Aids system versus the Standard system based on the above design considerations; Section 6.3 discusses the design alternative selected and basis for its selection; and Section 6.4 describes procedures for evaluation of each of the four aids individually to determine the conditions under which each is useful.

6.1 Design Considerations

In selecting an experimental design to be used for comparing the Aids system with the Standard system, several points had to be taken into consideration. First, of course, the design had to provide an appropriate method for comparison of the two systems in which human factors as well as measurement considerations had to be addressed. Shneiderman (47) has discussed some of the problems encountered in performing research involving human factors and presents three fundamental paradigms for such research—introspection, field studies, and controlled experiments. Although he feels that all three paradigms are useful, he indicates that studies involving controlled experiments will ultimately provide the most useful results. Since we agree with this analysis, we examined
various designs which could be used to evaluate the Aids system through controlled experiments.

Second, in order to perform a composite evaluation of the Aids system against the Standard system, some measure of quality of each system must be developed. Since the objective of the end user is to retrieve as many useful (relevant) documents as possible, while at the same time retrieving as few nonrelevant documents as possible, and since the final result of using either system is a search profile which is used to retrieve a set of reference documents, it is reasonable to use the quality of the retrieved document set as a measure of the profiling system which produced it. However, as discussed in Chapter 3, the relevance of each document, and the quality of a retrieved document set are both based on individual subjective judgements. Therefore, it is impossible to compare the quality of a document set retrieved by a given question with the quality of a different document set retrieved by a second question. Hence, it is desirable to use the same question in preparing a search request with both profiling systems so that it will be possible to evaluate which system performs better based on the relative quality of the two retrieved sets of documents as well as on the relative time needed to use each of the two systems.

Third, rather than trying to "artificially" gather users to test the system, it was desired to find users with a real information need. It was felt that this would provide better search requests and user attention for the experiment. The typical MIC patron is this type of user, and as such, the MIC patrons provided
a good pool of appropriate subjects. Unfortunately, this population of potential users did not have any experience using either the Standard system or the Aids system. Further, many of the users have had no interactive computer experience of any kind, and the main purpose of this research was to develop and evaluate the previously described aids for an existing state of the art system rather than to improve the user friendliness of such a system for casual users. Therefore, instead of trying to teach a large number of users how to use the two retrieval systems (and in many cases how to use a computer and/or how to do bibliographic searching), it was felt advisable to employ the assistance of an intermediary, despite the fact that, as discussed in Section 2.2., a debate is currently in progress as to the future of the intermediary for online searching. In this study, the use of intermediaries to aid in profile preparation should not be interpreted as an endorsement for either side in the above debate; rather, the choice was made based on the considerations presented above. Needless to say, the interactive aids developed in this research could prove useful, whether or not the intermediary is retained as an integral part of the search operation. (Research in the area of user friendly systems will determine the extent to which that may occur in the years ahead.)

Fourth, in order to utilize the MIC information specialists in accordance with the third consideration above, the experiment had to be designed so as not to seriously disrupt the normal operation of
MIC. This entire problem is discussed more fully in Section 6.3 in conjunction with the discussion of the design selection and its justification.

Finally, the costs and benefits to the user had to be considered. Although no monetary cost would be involved, some designs require the user to invest additional time without sufficient return in benefits to make him or her want to participate in the experiment. Consequently, one design alternative, which would have required the user to work with his profile in three different settings, was eliminated since it was felt that few MIC patrons would have participated in the experiment because of the excessive amount of time required.

6.2 Design Alternatives

Based on the above design considerations, two alternative evaluation designs were chosen as being the most appropriate for the current research although many others were possible. Each has advantages and disadvantages compared to the other, and for the potential benefit of anyone contemplating this type of research, both alternatives will be presented below. A discussion concerning the decision as to which alternative was actually employed will be given in Section 6.3.

6.2.1. Alternative 1 - Total User Involvement

The first design requires the end user to be involved throughout the entire process and would proceed as follows. The user and an intermediary will first code a profile using one system, and the
user and a different intermediary (this is discussed in more detail below) will then code a second profile for the same question using the other system. These two profiles will then be used to search a file of documents, and a set of documents will be retrieved for each of the profiles. The two sets will then be randomly mixed, and duplicates will be eliminated. The resulting combined set will be presented to the user, and he will judge each document in the set as to whether it is relevant to his question or not. At this point the user will not know which documents were retrieved by which profile. This is to eliminate any potential bias caused by the "likeability" of either system. After the relevance judgements have been made, the documents will be separated to form the two original sets.

The user will then be presented with the number of relevant and nonrelevant documents in each set and he will be asked which he prefers based on a five point scale (highly prefer set 1, slightly prefer set 1, no preference, slightly prefer set 2, highly prefer set 2). Again, the user will not be told which profile produced which set. Also, the evaluation data only will be presented rather than the actual list of documents. For example, the user will be asked, "Do you prefer Set 1 with 5 relevant documents and 33% precision (i.e. r=5,P=33%), or Set 2 with r=3 and P=50%?" This will be done so that one or two documents which "catch the user's fancy" do not unduly influence the user's judgement (since the results are based on a representative sample of documents and not the entire file as discussed in Section 7.1.1).
The above reflects the quality of results, but does not include the preference which might be expressed based on the time required to achieve the results. Unless the required time is substantially different, it may not affect user preference, but in certain cases it may have some bearing. For example, if the search results are identical, a score of "no preference" might be given based on output alone. However, if one profile took 30 minutes to construct while the other took 60 minutes, it is likely that the system which took only 30 minutes would be preferred.

In order to include this factor, the time to code each profile will be kept. After the user makes his preference selection based on output results only, he will then be asked questions to determine whether the time involved would affect his judgements. Again, so that the user cannot deduce which system generated which output, both times will be paired with both sets of output results. For example, the user will be asked, "If Set 1 took 30 minutes to obtain and Set 2 took 60 minutes, would your preference judgement change? Would it change if Set 1 took 60 minutes and Set 2 took 30 minutes?"

Only one of the above set-time pairings will be correct, and only the results from the correct pairing will be used in the evaluation. However, if the question is asked in this manner, the "likeability" bias will again be eliminated.

When several users have made the judgements discussed above, the two profiling systems can then be compared based on the proportion of users who favor the results obtained from the Aids
system versus the proportion of users who favor the results obtained from the Standard system.

As discussed above, for the current research, the regular patrons of MIC will be asked to participate in order to insure that an adequate number of users with real information needs are available. As also mentioned earlier, information specialists at MIC will be used to assist in profile preparation in order to eliminate the large amount of time to train the end user which otherwise would have been required. Further, any potential bias caused by the varying degrees of interactive computer experience of the end users is also eliminated by using the information specialists.

However, since the user will interact with the two systems in sequence, another source of bias which must be accounted for in the design is the learning effect which this will cause. In order to reduce the learning effect, a different specialist will be used to assist the user with each system; this, however, introduces another potential source of bias (the difference between information specialists) which must be accounted for in the design.

Therefore, 80 users (subjects) from the population of regular MIC patrons with internal Social Science search requests will be randomly selected to participate in the experiment. Each subject will be randomly assigned to either Group I or Group II, with each group containing 40 subjects. The subjects in Group I will use the Aids method first; the subjects in Group II will use the Standard method first. This will tend to balance any learning effect which might be present.
In addition, the information specialists assigned to work with each user will be determined randomly. Although MIC has four specialists available, scheduling does not permit all four to be available at all times. Therefore, the two specialists assigned to work with each user will be randomly selected from among all specialists available at the time the user is to go through the experimental procedure. Which of the two specialists will assist with the first system will also be randomly determined. This will randomize the effect caused by differences among information specialists.

Initially, the results can be evaluated using the following statistical model (see, for example, Lindgren (25) or Daniel (10)). Let \( x_i \) be the response from the \( i \)th subject.

\[
x_i = 0 \text{ if the Standard method is preferred at either level}
\]
\[
1 \text{ if the Aids method is preferred at either level}
\]

(A score of "no preference" will be discarded.)

Then,

\[
\hat{p} = \frac{1}{N} \sum_{i \in I} x_i
\]

(6.1)

where \( N \) is the number of responses which show a preference for either system,

and \( \hat{p} \) is an estimate of the binomial proportion of the population which favors the Aids system over the Standard system.

We can use \( \hat{p} \) to test the following hypothesis.

\[ H_0: p \leq \frac{1}{2} \quad \text{(Standard favored or no difference)} \]

\[ H_1: p > \frac{1}{2} \quad \text{(Aids favored)} \]
This test assumes that if the Aids system is providing no help to the user, approximately half of the users will favor each system, with preferences caused by individual user questions and views being divided equally between the two systems. However, if \( p \) is significantly greater than \( \frac{1}{2} \), it will support the argument that the Aids system is performing a service to at least some users that they cannot obtain from the Standard system.

Since 80 subjects are being tested for the current experiment, unless there are a substantial number of "no preference" judgements, \( N \) will be large enough\(^a\) that the normal approximation for the binomial test can be used. Thus, the statistic \( z \) can be computed from Equation 6.2, and a table of the standard normal distribution can be used to determine the significance of the obtained value of \( z \). The test will be applied first to the results based on the output only, and second to the results based on both output and time required to code the profile.

\[
z = \frac{p - p_o}{\sqrt{\frac{p_o(1-p_o)}{N}}} = \frac{p - .5}{\sqrt{\frac{.25}{N}}} \tag{6.2}
\]

There are at least two potential problems with the above procedure. First, it may turn out that there are many "no preference" responses. In this case, the above model may not be useful. However, the large number of "no preference" responses may indicate that the Aids system is only marginally useful. An examination of the individual aids (see below) should help determine if this is indeed true.

\(^a\)A value of \( N=10 \) is considered to be large enough if \( p=.5 \). If \( p \) is very close to zero or one, then \( N \) must be much larger. For the anticipated \( p \) value for the population studied, a value \( N=20 \) should be more than sufficient (48).
A second potential problem is that it seems on initial examination that the system used second will profit from a learning effect as discussed earlier. If the learning effect turns out to be quite strong, the system used second may win the preference vote in almost all cases. Should this happen, the original five point scale may provide some justification for the overall performance of one system or the other. For example, if System A is strongly preferred quite often when used second, but System B is always preferred only slightly there is some justification for a contention that System A is superior overall. Again, an examination of the individual aids should help in making this determination.

6.2.2. Alternative 2 - Limited User Involvement

The second potential design procedure is somewhat similar, but in this design the user is only involved to a limited extent. The same population of users is involved, but the user would simply go through the standard MIC interview process and then his participation would end. At this point one information specialist would enter and manipulate the profile using one system, while another specialist would enter and manipulate the profile using the other system. This design eliminates the bias caused by the learning effect since the profiles would be constructed independently. However, the effect caused by the differences among specialists might be more pronounced, so again, the assignment of system to specialist will be randomized in order to randomize the specialist effect.

As discussed in Section 6.3, the first alternative was really the design of choice; this alternative was a backup alternative imposed by operational constraints discussed in Section 6.3.
The evaluation in this design will then be performed by the pair of specialists who coded the search. It would proceed as indicated in the first design, with the exception that if the specialists could not agree on the relevance of the document, it would be recorded as a "don't know", and that document would not be counted regardless of which profile produced the hit. This is done to eliminate any error effect in the results caused by the specialists disagreeing over the actual needs of the user.

At this point a note should be made concerning the evaluation of relevance by a pair of specialists rather than the end user. As discussed in Chapter 3, the end user is the ultimate judge of relevance for his search. In this design, however, despite the understanding reached with the user during the patron interview, further questions concerning the user's information need often arise during an online search. Since this design does not provide for interaction with the user during the online search, the specialist at times must make a decision concerning the direction of the search, and this decision may or may not correspond to the decision which would have been made with the user present. Thus, rather than introducing potential bias caused by one specialist guessing correctly concerning the user's needs, but the other specialist guessing incorrectly, it was felt that evaluation by the specialists as discussed above (rather than by the end user) would be more appropriate for this design.

After the relevance judgements are made, the choice of preferred system would be made using the five point scale and other procedures
described for the last design. The only difference would be that, as with the relevance judgements, the pair of specialists who coded the search would make the choice of system preference. Although the choice of system would be obvious in many cases (see Chapter 3), should the specialists disagree as to which set of results is preferred, or should one specialist record a vote of "no preference", a score of "no preference" would be recorded. Should one specialist highly favor one set of results while the other specialist slightly favors the same set of results, a score of "slightly prefers" the system which generated the preferred set would be recorded.

After the preference scores have been recorded for all searches, the statistical evaluation would then be conducted exactly as described for the other design.

6.3 Design Alternative Chosen for the Experiment

The first design alternative (Section 6.2.1) was originally considered to be the design of choice because of more end-user involvement. However, the second alternative was chosen, somewhat reluctantly, for the following pragmatic reasons in order to complete the experiment in a reasonable period of time.

When the design alternatives were originally discussed with the MIC staff, they felt that Alternative 1 could be worked into their operational schedule without causing a major disruption. However, since the original planning meeting there have been fairly severe budget cutbacks throughout the University as a result of
which MIC was forced to give up an information specialist position until the University's economic position improved.

Even with this development and the resultant work overload placed on the remaining information specialists, the MIC staff felt that the first design alternative could still be used if the experimenter could participate as one of the two information specialists required for each patron, and if the data gathering could be limited to avoid certain weeks during the quarter when the demand for MIC services was at its highest. However, despite the willingness and help of the information specialists, the first design alternative proved to be too cumbersome from an operational standpoint because the MIC specialist could not afford to spend time working online with a patron when the experiment dictated. This was especially true if the trial dictated that the experimenter were to work with the patron first. The waiting period that this entailed for the MIC information specialist followed by the time then required to interact with the patron online would have degraded service to other patrons to an unacceptable level. Therefore, when the pilot study for the first design alternative brought the operational constraints into clear focus, the second design alternative was chosen out of necessity.

Trials with the second design proceeded smoothly, and in only a few instances did the specialists disagree as to the relevance of the documents retrieved. Also, the disagreements tended to be of the form where one specialist felt a document was "probably
relevant, but not sure", while the other specialist felt it was "probably not relevant, but not sure." As indicated in Section 6.2.1, these documents were judged "don't know", but they probably would have been judged "don't know" even if only one specialist had been doing the evaluation. This low incidence of disagreement can possibly be attributed to the fact that great care was taken to ensure that the specialists had a common understanding of the user's needs. This was accomplished primarily by conducting a joint interview with the patron. In addition, a brief conversation to clarify understanding between specialists occurred when necessary. Thus, while the specialists may not have evaluated the output exactly as the user might have evaluated it, they did make the evaluation based on the information that they were attempting to retrieve from the database based on their understanding of what the user wanted.

One favorable trade off for the second design alternative, which helped to compensate for the loss of user participation in relevance judgements and system preference judgements, was that it eliminated any bias caused by the learning effect discussed in Section 6.2.1 for design alternative 1. Additional details concerning the actual implementation of design 2 are presented in Chapter 7 together with the experimental results.

6.4 Evaluation of the Individual Aids

The above discussions deal with design considerations for composite evaluation of the Aids system against a baseline, state
of the art online retrieval system. This type of evaluation, however valuable, will provide little information concerning the functioning of the individual aids. Even if the Aids system is found to be superior to the Standard system, one or more of the individual aids may prove to contribute little to the effectiveness of the overall system. In addition, due to the diverse nature of the aids, one aid might function well under some conditions, while another might function well under other conditions. Therefore, procedures for evaluating the individual aids would also be desirable.

Evaluation of the individual aids will be performed by examining the log of each profiling session in which the Aids system was used. From the log, several pieces of information will be collected for each aid. First, the coverage or applicability of an aid can be determined by the number of times it appears. For example, from the usage statistics for the profile-profile association aid it will be possible and useful to note how often a profile similar to the new profile exists in the previous profile set. If this is an infrequent, but very useful occurrence, a good case can be made for expanding the set of existing profiles used for the profile-profile association aid. A second measure will determine the number of times that a profiling suggestion made by an aid is accepted by the user. A large number of irrelevant profiling suggestions would be not only useless but also annoying, and may suggest that the display thresholds for the various association lists discussed in Chapter 5
should be modified. A final measure will determine the number of times that a profiling suggestion made by an aid is unnecessary because the user had thought of the suggestion on his own. This measure could provide guidelines as to when the aids should be invoked relative to the expertise level of the user. While no tests of hypothesis will be performed on the individual aids, the descriptive statistics presented above will provide good insight into how each individual aid is functioning.
CHAPTER VII. EXPERIMENTAL RESULTS

This chapter will provide an analysis of the actual data gathered during the course of this research by using the techniques and experimental design discussed in the last chapter. Specific comments will be provided as deemed necessary. General inferences and conclusions which seem warranted after examining the data and its analysis are presented in the concluding chapter.

7.1 The Evaluation of the Aids System

The general experimental design for this evaluation was presented in the last chapter. The important points will be repeated below, and, of course, the actual data will be included.

7.1.1 Gathering of the Evaluation Data

The population used in this study is that population of normal MIC patrons who approached MIC during the data gathering period to request a search of the internal Social Sciences database. Also, the population was further limited to include only those patrons who were willing to participate in the joint interview with both an MIC information specialist and the experimenter present.

The interview was conducted in the normal manner by the information specialist with the exception that the experimenter added any comments or suggestions which he felt were relevant to the coding process at hand. The coding strategy was written directly on an MIC coding form or sketched out in note form depending on the type of search and the preference of the information specialist. If the strategy were sketched in note form, it was
then written on the coding sheet (possibly in a slightly modified form after a term frequency list had been consulted) as soon as time permitted. The coding sheet was then used as a common starting point from which the two online searches for each search request were conducted. If any points of confusion were discovered, these were clarified by mutual agreement of the searchers before proceeding with the online sessions.

The experimenter participated in every trial. The number of trials in which each information specialist participated was determined by the number of patrons in the population who were interviewed by the particular information specialist. This served to randomize the specialist who participated with the experimenter in each trial. Since the experimenter did not have the day to day experience in conducting searches which the information specialists had, it was felt necessary to account for this potential source of bias by balancing the trials with respect to the system used. Thus, for the 80 trials, the experimenter used the Standard system for 40 trials and the Aids system for the other 40 trials.

The online sessions were conducted against the learning file of 24,878 documents (see Section 5.4) at a time convenient to the searchers and as soon as possible after the interview. As each online search was completed, the resulting profile was stored in an internal form in the computer system. The evaluation program then used the two stored profiles for each search request to search a test file similar to the learning file used to construct
the profiles. This test file also represented approximately 5% of
the MIC Social Science retrospective database and was constructed
in a manner similar to the learning file, but consisted of a different
5% sample of the database.

The bibliographic references retrieved from each pair of
profiles were sorted internally into three groups: those documents
retrieved by the Standard system only, those documents retrieved
by the Aids system only, and those documents retrieved by both
systems. The documents were shuffled internally before display
so that their group membership was unknown to the evaluators (the
experimenter and the specialist who had coded the search). Each
document was judged to be either relevant, nonrelevant, or of
unknown relevance by the evaluators. The relevance of a document
was designated as unknown if the searchers could not agree on
its relevance, or if neither could make an accurate judgment (such
as with a foreign language title). Such a document was not
included in the final tabulated results.

After the individual documents had been evaluated, the results
for each system were compiled and then displayed in the format:

ONE PROFILE HAS X RELEVANT WITH A PRECISION OF Y

THE OTHER PROFILE HAS U RELEVANT WITH A PRECISION OF V

The searchers then scored the results on the five point scale dis­
cussed in Sections 6.2.1 and 6.2.2. After the results had been
recorded, the program indicated whether the Aids or Standard
results had been displayed first (the order was randomized
internally). The results of the searches, the judgments as to
which search system was preferred, and relevant comments are presented in detail in Table 7.1. An overall summary of the evaluative results is presented in Table 7.2.

### 7.1.2 Statistical Tests for Significance

From Table 7.1 and Table 7.2, it can be seen that in 29 cases the profile coded using the Aids system was preferred to the profile coded using the Standard system. The opposite was true in only eight cases, while no preference (i.e., a tie) was found in 43 cases. Thus, applying the statistical test described in Section 6.2.1 (excluding ties), we obtain from equations 6.1 and 6.2, respectively

\[
\hat{p} = \frac{1}{N} \sum_{i=1}^{k} x_i = \frac{29}{37} = .78
\]

\[
z = \frac{\hat{p} - p_0}{\sqrt{\frac{p_0(1-p_0)}{N}}} = \frac{.78 - .50}{\sqrt{\frac{.25}{37}}} = 3.45
\]

The results, summarized in Table 7.3, show that the Aids system is preferred at the 0.01 significance level.

Due to the relatively large number of ties (see Section 7.1.3 for further discussion), two other statistical tests were considered: a Distribution-Free Test based on Friedman Rank Sums and the Fisher Sign Test (both are described in Hollander and Wolfe (17)). Although the Friedman Rank Sum Test provides a specific correction for ties, Hollander and Wolfe (17) note that for only two treatments (which is the case for this experiment) the Friedman Rank Sum Test reduces to the Fisher Sign Test. In the procedure for the Fisher
Table 7.1

Evaluation Data for Comparison of the Standard System with the Aids System

<table>
<thead>
<tr>
<th>Profile Number</th>
<th>Standard System Profile</th>
<th>Aids System Profile</th>
<th>Judgment</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>a</td>
<td>b</td>
<td>a</td>
<td>b</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>50</td>
<td>4</td>
<td>50</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>9</td>
<td>1</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>67</td>
<td>3</td>
<td>75</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>33</td>
<td>2</td>
<td>50</td>
</tr>
<tr>
<td>5</td>
<td>18</td>
<td>90</td>
<td>18</td>
<td>90</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>27</td>
<td>0/0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>0/0</td>
<td>0</td>
<td>0/0</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>0/3</td>
<td>0</td>
<td>0/2</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>0/5</td>
<td>0</td>
<td>0/5</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>0/2</td>
<td>0</td>
<td>1/3</td>
<td>1</td>
</tr>
<tr>
<td>11</td>
<td>18</td>
<td>75</td>
<td>18</td>
<td>75</td>
</tr>
<tr>
<td>12</td>
<td>4</td>
<td>36</td>
<td>4</td>
<td>36</td>
</tr>
<tr>
<td>13</td>
<td>1</td>
<td>100</td>
<td>1</td>
<td>100</td>
</tr>
<tr>
<td>14</td>
<td>1</td>
<td>13</td>
<td>1</td>
<td>13</td>
</tr>
<tr>
<td>15</td>
<td>0/0</td>
<td>0</td>
<td>0/0</td>
<td>0</td>
</tr>
<tr>
<td>16</td>
<td>7</td>
<td>70</td>
<td>1</td>
<td>70</td>
</tr>
<tr>
<td>17</td>
<td>2</td>
<td>100</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>18</td>
<td>3</td>
<td>100</td>
<td>3</td>
<td>100</td>
</tr>
<tr>
<td>19</td>
<td>0/1</td>
<td>0</td>
<td>0/1</td>
<td>0</td>
</tr>
<tr>
<td>20</td>
<td>4</td>
<td>80</td>
<td>4</td>
<td>57</td>
</tr>
<tr>
<td>21</td>
<td>0/2</td>
<td>0</td>
<td>0/2</td>
<td>0</td>
</tr>
<tr>
<td>22</td>
<td>4</td>
<td>80</td>
<td>4</td>
<td>80</td>
</tr>
<tr>
<td>23</td>
<td>2</td>
<td>40</td>
<td>2</td>
<td>33</td>
</tr>
<tr>
<td>24</td>
<td>6</td>
<td>100</td>
<td>8</td>
<td>100</td>
</tr>
<tr>
<td>25</td>
<td>4</td>
<td>57</td>
<td>4</td>
<td>57</td>
</tr>
<tr>
<td>26</td>
<td>7</td>
<td>88</td>
<td>8</td>
<td>67</td>
</tr>
<tr>
<td>27</td>
<td>0/2</td>
<td>0</td>
<td>0/2</td>
<td>0</td>
</tr>
<tr>
<td>28</td>
<td>11</td>
<td>69</td>
<td>11</td>
<td>73</td>
</tr>
<tr>
<td>29</td>
<td>2</td>
<td>67</td>
<td>2</td>
<td>67</td>
</tr>
<tr>
<td>30</td>
<td>5</td>
<td>83</td>
<td>9</td>
<td>90</td>
</tr>
<tr>
<td>31</td>
<td>0/2</td>
<td>0</td>
<td>0/4</td>
<td>0</td>
</tr>
<tr>
<td>32</td>
<td>1</td>
<td>7</td>
<td>1</td>
<td>7</td>
</tr>
<tr>
<td>33</td>
<td>4</td>
<td>36</td>
<td>4</td>
<td>36</td>
</tr>
<tr>
<td>34</td>
<td>1</td>
<td>25</td>
<td>2</td>
<td>40</td>
</tr>
<tr>
<td>35</td>
<td>3</td>
<td>75</td>
<td>5</td>
<td>42</td>
</tr>
<tr>
<td>36</td>
<td>9</td>
<td>100</td>
<td>10</td>
<td>100</td>
</tr>
<tr>
<td>37</td>
<td>2</td>
<td>33</td>
<td>2</td>
<td>33</td>
</tr>
<tr>
<td>38</td>
<td>3</td>
<td>67</td>
<td>3</td>
<td>67</td>
</tr>
<tr>
<td>39</td>
<td>7</td>
<td>70</td>
<td>7</td>
<td>70</td>
</tr>
<tr>
<td>40</td>
<td>6</td>
<td>86</td>
<td>6</td>
<td>100</td>
</tr>
</tbody>
</table>

a: r is the number of relevant retrieved documents
b: P is the search precision (percent)
c: Judgment Values are defined as follows:
   1. Aids Highly Favored
   2. Aids Slightly Favored
   3. No Preference
   4. Standard Slightly Favored
   5. Standard Highly Favored
d: The Comments are defined as follows:
   1. No modification was attempted
   2. Aids suggested new terms which were incorporated by the specialist
   3. Terms suggested by Aids and incorporated did not retrieve new documents in test sample
   4. Terms suggested by Aids and incorporated retrieved new documents which were irrelevant
   5. Aids search was modified by specialist based on retrieval counts or sample documents
   6. Standard search was modified by specialist based on retrieval counts or sample documents
   7. Specialist using Aids system first included new terms suggested by Aids but decided against their inclusion after examining some sample documents
   8. In the case where P = 0%, the total number retrieved is also shown
Table 7.2

A Summary of the Evaluation Data From Table 7.1

<table>
<thead>
<tr>
<th>Data Point</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of trials</td>
<td>80</td>
</tr>
<tr>
<td>Number of &quot;no preference&quot; judgments (ties)</td>
<td>43</td>
</tr>
<tr>
<td>Number of times the search results from the profile coded using the Aids system were preferred</td>
<td>29</td>
</tr>
<tr>
<td>Number of times the search results from the profile coded using the Standard system were preferred</td>
<td>8</td>
</tr>
</tbody>
</table>

Table 7.3

Summary of the Statistical Test for Significance
Excluding Ties\(^a\)

<table>
<thead>
<tr>
<th>Variable (^b)</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>37</td>
</tr>
<tr>
<td>(\hat{p} = \frac{1}{N} \sum_{i=1}^{N} x_i = \frac{29}{37})</td>
<td>0.78</td>
</tr>
<tr>
<td>(z = \frac{\hat{p} - p_0}{\sqrt{\frac{p_0(1-p_0)}{N}}} = \frac{0.78 - 0.50}{\sqrt{\frac{0.25}{37}}} = \frac{0.28}{0.082} = 3.45) (^c)</td>
<td>3.45</td>
</tr>
</tbody>
</table>

\(^a\) Using summary data from Table 7.2

\(^b\) See Section 6.2.1

\(^c\) Significant at the 0.01 level for the Hypothesis

\(H_1: p > \frac{1}{2}\) (Aids system favored)
Sign Test, any ties are discarded and the number of trials is redefined to be the number of trials which are not tied (that is what was done above). Further, the large sample approximation for the Fisher Sign Test computes a statistic which is compared to the standard normal distribution. The statistic computed for the Fisher Test mathematically reduces to the statistic computed above. Thus, in this case, the two tests yield identical results.

7.1.3 Analysis and Treatment of Ties

The test described in the last section has indicated that the Aids system is preferred to the Standard system in a (statistically) significant number of cases, if ties are discounted. However, the large number of ties encountered in the evaluative study cannot be totally dismissed in any practical analysis of the results.

One method of viewing the ties is as follows. If we assume that every user does in fact prefer either the Standard system or the Aids system, then we can conclude that the conditions under which the evaluation study was performed may have prohibited a discrimination between the two systems in certain cases. (Experimental conditions which may have affected the evaluative results thusly are discussed below and in the next section.) Thus, the ties may be viewed as "noise" data points, and since it seems reasonable to conclude that each tie is just as likely to favor the Aids system as to favor the Standard system, it is logical to assign an $x_i$ value of 1/2 in Equation 6.1 for each "no preference" case recorded in Tables 7.1 and 7.2. Under this treatment, a large enough noise group will force the
value of $\tilde{p}$ (Equation 6.1) toward 0.5 with the desired effect that the statistical test using equation 6.2 in Section 6.2.1 will now reflect the effect of a sizable number of ties. The results from this treatment, shown in Table 7.4, still indicate that the Aids system is preferred at the 0.01 significance level.

As mentioned above, one of the assumptions made in the above treatment of ties suggests that the conditions under which the evaluative study was performed may have precluded adequate discrimination between the Aids system and the Standard system in certain cases. An additional basis for discrimination would have been available in the original experimental design (Section 6.2.1), where the evaluations were to have been based in part on the length of time required to code the profile interactively with each system. Section 7.1.4 below explains why time measurements could not be used for the conditions under which the evaluation studies were actually performed, and also explains why many of the profiles would have been expected to be coded in less time with the Aids system than with the Standard system. Thus, as suggested by the evidence presented in both the next section and Section 7.2, many of the ties recorded under the conditions of the experiment most likely would have been broken in favor of the Aids system if the experiment had been performed as originally planned. Repetition of the experiment under the original design conditions would help to verify or refute this claim.

7.1.4 Profile Preparation Time Considerations

As mentioned above, in the original experimental design for the evaluation study (Section 6.2.1), the comparative evaluation of the
Table 7.4

Summary of the Statistical Test for Significance
Including Ties

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>80</td>
</tr>
</tbody>
</table>

\[ \hat{p} = \frac{1}{N} \sum_{i=1}^{N} x_i = \frac{50.5}{80} = 0.63 \]

\[ z = \frac{\hat{p} - p_o}{\sqrt{\frac{p_o(1-p_o)}{N}}} = \frac{0.63 - 0.50}{\sqrt{\frac{0.25}{80}}} = \frac{0.13}{0.056} = 2.33 \]

---

a Using summary data from Table 7.2
b See Section 6.2.1 equations 6.1 and 6.2
c
\[ x_i = \begin{cases} 
0 & \text{if the Standard system is preferred} \\
1 & \text{if the Aids system is preferred} \\
\frac{1}{2} & \text{if there is no preference} 
\end{cases} \]
d Significant at the 0.01 level for the Hypothesis
\[ H_1: \ p > \frac{1}{2} \] (Aids system favored)
two systems was to have been based in part on the length of time required to code the profile interactively with each system. Because of the necessity to use the alternative design (Section 6.2.2), a good deal of the time involved in coding the profile was spent during the patron interview. Although the interview time was not specifically measured, it averaged 15-20 minutes, and generally provided an excellent "starting" profile. This extensive interview process resulted in greatly reducing the required online time and favored the Standard system. For example, if the design discussed in Section 6.2.1 had been used, the interview could have been very brief, and the strategy constructed as the information specialist and user explored the database together. In this case, the "think" time required during either the preliminary interview or the interactive session using the Standard system would have been reduced by the suggestions provided by the Aids system. This contention is supported by the fact that the Aids system suggested many terms already on the coding sheet. (This is discussed more fully in Section 7.2 and summarized in Tables 7.5 and 7.6.) In addition, if searching by the end user without the assistance of an intermediary is anticipated, the online time for both systems should increase significantly, and the guidance provided to a casual user by the Aids system should make the Aids system more time efficient in most cases than the Standard system. Again, repetition of the experiment under the original design conditions would help to verify or refute this claim.

In view of the above, it is felt that the time measurements recorded during this experiment are of little value. The average time spent when using the Aids system was slightly over nine minutes, while the average time spent using the Standard system was 3 1/2 minutes.
The time difference is largely accounted for by the time spent displaying the various aids on the relatively slow hard copy terminal used to conduct this research, and as discussed in Section 7.2, under the conditions of the experiment, the aids usually provided terms which were already on the coding sheets.

7.1.5 Computer Overhead Considerations

Thus far, the evaluation of the two systems being compared has centered on the effect of the user. However, the effect on the computer system should also be mentioned. To begin with installation of a standard online retrieval system requires a large commitment of computer resources. In particular, a large amount of disk storage is required to hold both the index file and the file of document references. Additionally, the computer time spent searching these disk files can be significant for even a moderate number of users. Furthermore, if any of the aids discussed in this dissertation are to be added to a standard online retrieval system, both additional disk space and additional execution time will be required.

The amount of additional disk storage required to install the Aids system as compared to the Standard system can be estimated by examination of Appendix B. However, from this study it is difficult to estimate the difference in computer time required for the two systems, since the normal retrieval environment entails multiple users interacting with the retrieval software in an environment where the computer is dedicated to the retrieval function, while this study was conducted in a general timesharing environment with only one user interacting with the retrieval software at a given time. Under these
conditions there was no apparent difference in response time between the Aids system and the Standard system. It should be noted that the additional amount of disk storage required for the Aids system can be reduced at the usual expense of additional processing time, if that can be tolerated.

7.1.6 Impact of Omitted State of the Art Features on the Standard and Aids Systems

As mentioned in Section 4.3, there are three state of the art features not included in either of the systems used in this study whose impact on this research will now be discussed. First, there is no command which allows the user to browse through the inverted file index. This omission would tend to cloud the results of the study if many of the terms suggested by the aids were simply variant word forms of the initial terms which were entered. It could then be argued that a simple index browsing command would bring the Standard almost to a par with the Aids system. However, this did not prove to be the case. Although there were some cases where the aids suggested alphabetically close terms (in addition to the more preponderant non-alphabetically close terms), in almost all such cases these words were already included in the search strategy. Therefore, this omission had only minor impact on the results of the study.

The second missing feature, not displaying all terms retrieved by a truncated term, was compensated for through use of an index term frequency listing by the MIC information specialists before beginning the online search. This, coupled with the experience of the information specialists, prevented problems of over-truncation just as well as if
The omitted feature had been available.

The final omission, lack of postings display for the individual terms when several are entered on one line, could be circumvented, if the searcher felt it desirable, by entering the search terms one per line to obtain postings for the individual terms and then combining the terms using the associated line numbers. Therefore, this omission caused an inconvenience only with no loss of capability.

7.1.7 Comparison of Observed System Preferences With Those Predicted by the Utility Model

A comparison of the system preference judgments actually made by the information specialists (see Table 7.1) with those predicted by the utility model presented in Section 3.4, using the utility graph shown in Figure 3.4 yields the following results. For the 43 cases where the observed judgment was "no preference," the utility graph would have predicted the same judgment for 42 of them. In the other case (#64), the graph would have given a slight edge to the Aids system (r=1, P=33% over r=2, P=8%). For the 37 cases where the specialists indicated a preference for one system or the other, 29 of these were "clear-cut" cases as described in Section 3.2 (both r and P better or one better with the other equivalent). In all 29 "clear-cut" cases both the specialists and the utility graph favored the system which produced superior results. In the remaining eight cases where the information specialists indicated a preference (#26, #35, #38, #46, #60, #68, #72, and #76), the r value was higher for one system while the P value was higher for the other system. Comparing the results it can be seen that for all of these cases, except #26 and #60, the
judgement of the specialists corresponded to that predicted by the utility graph. For example, for profile #38 with Standard results \( r=2, P=67\% \) compared with Aids results \( r=4, P=36\% \), the specialists chose the results produced by the Aids system. The graph puts \( r=2, P=67\% \) slightly below the line with rank 2, while it puts \( r=4, P=36\% \) about half way between the lines with rank 2 and 3. Thus, the graph would favor the \( r=4, P=36\% \) results also. For profiles #26 and #60, where the specialists slightly favored the Aids system, the utility graph would have predicted no preference in each case. The extremely high correspondence between the observed evaluation judgments and those predicted by the utility model is very encouraging, and the significance is discussed in the next chapter.

7.2 Evaluation of the Individual Aids

Although the Aids system as a whole was judged to be more useful than the Standard system in a statistically significant number of cases, it is also desirable to evaluate the usefulness of the individual aids. As can be seen from the summaries in Tables 7.5 and 7.6, all three association aids proved to be useful. The performance of the two term-term association aids is summarized on a term efficiency basis in Table 7.5 and the performance of all three association aids is summarized on a profile efficiency basis in Table 7.6.

As shown in Table 7.5, 57\% of the lists generated by the term-term (profile) aid and 55\% of the lists generated by the term-term (document) aid produced either new useful terms or terms already on the coding sheet. While generation of terms already on the coding sheet provided no new ideas for profile development in this experiment, such would not have been the case had the coding prior to the start of the
Table 7.5

Performance of the Term-Term Association Aids on a Term Efficiency Basis

<table>
<thead>
<tr>
<th>Aid</th>
<th>Criterion</th>
<th>Result&lt;sup&gt;a&lt;/sup&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) T-T(P)&lt;sup&gt;b&lt;/sup&gt;</td>
<td>(1.1) Number (and percent) of Profile Terms generating a list</td>
<td>457 (52%)</td>
</tr>
<tr>
<td></td>
<td>(1.2) Number (and percent) of lists contributing new useful profile terms</td>
<td>49 (11%)</td>
</tr>
<tr>
<td></td>
<td>(1.3) Number (and percent) of lists containing terms already on profile coding sheet</td>
<td>229 (50%)</td>
</tr>
<tr>
<td></td>
<td>(1.4) Number (and percent) of lists containing useful terms (union of 1.2 and 1.3)</td>
<td>258 (57%)</td>
</tr>
<tr>
<td>(2) T-T(D)&lt;sup&gt;c&lt;/sup&gt;</td>
<td>(2.1) Number (and percent) of Profile Terms generating a list</td>
<td>723 (82%)</td>
</tr>
<tr>
<td></td>
<td>(2.2) Number (and percent) of lists contributing new useful profile terms</td>
<td>90 (12%)</td>
</tr>
<tr>
<td></td>
<td>(2.3) Number (and percent) of lists containing terms already on profile coding sheet</td>
<td>357 (49%)</td>
</tr>
<tr>
<td></td>
<td>(2.4) Number (and percent) of lists containing useful terms (union of 2.2 and 2.2)</td>
<td>398 (55%)</td>
</tr>
</tbody>
</table>

<sup>a</sup> Based on a total of 880 unique terms in the 80 resultant profiles

<sup>b</sup> Term-Term (Profile) Association

<sup>c</sup> Term-Term (Document) Association
Table 7.6

Performance of the Individual Aids on a Profile Efficiency Basis

<table>
<thead>
<tr>
<th>Aid</th>
<th>Criterion</th>
<th>Result$^a$</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) $T-T(P)^b$</td>
<td>(1.1) Number (and percent) of profiles generating lists</td>
<td>80 (100%)</td>
</tr>
<tr>
<td></td>
<td>(1.2) Number (and percent) of profiles for which lists contributed new useful profile terms</td>
<td>29 (36%)</td>
</tr>
<tr>
<td></td>
<td>(1.3) Number (and percent) of profiles for which lists contained terms already on profile coding sheets</td>
<td>61 (76%)</td>
</tr>
<tr>
<td></td>
<td>(1.4) Number (and percent) of profiles for which lists contained useful terms (union of 1.2 and 1.3)</td>
<td>65 (81%)</td>
</tr>
<tr>
<td>(2) $T-T(D)^c$</td>
<td>(2.1) Number (and percent) of profiles generating lists</td>
<td>80 (100%)</td>
</tr>
<tr>
<td></td>
<td>(2.2) Number (and percent) of profiles for which lists contributed new useful profile terms</td>
<td>57 (71%)</td>
</tr>
<tr>
<td></td>
<td>(2.3) Number (and percent) of profiles for which lists contained terms already on profile coding sheets</td>
<td>72 (90%)</td>
</tr>
<tr>
<td></td>
<td>(2.4) Number (and percent) of profiles for which lists contained useful terms (union of 2.2 and 2.3)</td>
<td>76 (95%)</td>
</tr>
<tr>
<td>(3) $P-P^d$</td>
<td>(3.1) Number (and percent) of times a similar profile was found</td>
<td>12 (15%)</td>
</tr>
<tr>
<td>(4) Composite</td>
<td>(4.1) Number (and percent) of profiles for which potentially useful terms were generated (union of 1.4 and 2.4)</td>
<td>78 (97%)</td>
</tr>
<tr>
<td></td>
<td>(4.2) Number (and percent) of profiles for which potentially useful terms were generated or similar profiles were found (union of 1.4, 2.4, and 3.1)</td>
<td>78 (97%)</td>
</tr>
</tbody>
</table>

$^a$ Based on a total of 80 profiles

$^b$ Term-Term (Profile) Association

$^c$ Term-Term (Document) Association

$^d$ Profile-Profile Association
interactive session not been so comprehensive. Hence, as mentioned in Section 7.1.4 the terms produced by all of the association aids would have been much more useful and the total amount of time required to develop a suitable profile probably would have been reduced had the interactive session been initiated at the start of the interview with each patron. This inference appears to be further substantiated by the results summarized in Table 7.6 which show that potentially useful terms were generated (or similar profiles found) for 97% of the profiles. However, repetition of the evaluation studies under the original design conditions described in Section 6.2.1 would help to verify or refute the purported usefulness of the association aids.

With regard to the profile-profile association aid, as shown in Table 7.6, similar profiles were found in only 12 of the 80 cases. Even though these profiles proved to be useful in only a few cases (again because of the prior coding problem) the results are encouraging considering the overall likelihood of a new query being exactly identical to an earlier query. As discussed more fully in the next chapter, the value of this aid may have been diminished by the relatively small number of profiles in the set.

The relevance feedback aid, although automatically invoked for most of the searches performed with the Aids system, proved to be of real value in only two or three cases, and of moderate value in six or seven other cases. These results are also encouraging considering the conditions under which the study was performed. Conditions under which this aid would prove to be more useful are discussed more fully in the next chapter.
CHAPTER VIII. CONCLUSIONS AND DIRECTIONS FOR FUTURE RESEARCH

This research was oriented toward the development and evaluation of online, interactive aids to help a user adequately specify his search topic when querying a document retrieval system. The four interactive aids and the Aids system which were developed and implemented for this purpose were evaluated individually and compositionally. The evaluative results were presented in Chapter 7, while the conclusions to be drawn and suggestions for future research are presented below.

The composite evaluation showed that, for interactive profile preparation, the Aids system was preferred over the Standard system in a statistically significant number of cases. The relatively large number of cases in which there was no preference (i.e., a tie) between the two systems was somewhat unexpected. However, as discussed in Section 7.1.3, had it been possible to employ the original design methodology described in Section 6.2.1, many of the ties probably would have been broken in favor of the Aids system because the Aids system should require less "think" time to develop search profiles from scratch than would the Standard system. The backup design methodology used in this study (necessitated by operational constraints), caused the online time measurements to be rather meaningless and precluded the use of such measurements in the evaluative study. Repetition of the experiment using the original design methodology, when and if the operational environment permits, would help to verify
whether or not the Aids system would in fact require less time than the Standard system to develop a profile from scratch. Such a study would also involve the end user in a more meaningful way, as was originally intended.

Regarding the individual aids, the three association aids worked surprisingly well. The term-term association aid based on previous profiles generated more association lists than anticipated. Lists were generated for 52% of the profile terms entered interactively, and the lists provided useful terms over 50% of the time. The term-term association aid based on the sample document set generated a greater number of lists, as anticipated. Lists were generated for 82% of the profile terms entered interactively and, somewhat surprisingly, these lists also provided useful terms over 50% of the time. The fact that the terms on these lists were more useful than anticipated (see Section 2.6 for discussion) is probably due to the large number of MARC records contained in the Social Science database (of which the learning file used to construct the aids was a subset). In addition to title words, the MARC records contain LC subject headings and descriptors. The association measure thus provided relationships between words in the titles and words in the subject headings and vice versa. Also, the words in the subject headings were associated to each other. This produced an effect much like a controlled vocabulary, and surely produced better association lists than would have been produced by title words alone.

The profile-profile association aid found similar profiles in 12 of the 80 cases. These results are encouraging considering the overall
likelihood of a new query being exactly identical to an earlier query. Also, the chance of finding a similar profile was further diminished by the relatively small set of profiles available for this study\(^a\) (342) and the relatively high threshold (.70) needed to activate the aid. By using such a high threshold, it was thought that when a similar profile was found it would be of use to the profile coding process. It was disappointing to find that in many cases earlier profiles retrieved by this aid were of limited use to the coding process. This, too, was probably a result of the high quality of starting profile produced by the methodology employed in this experiment. Again, an experiment using the original design methodology together with a larger set of earlier profiles\(^a\) would help determine the true usefulness of this aid.

The relevance feedback aid did not perform as well as hoped because of two basic reasons. First, the set of relevant documents with which this aid was invoked in this study was relatively small (4-10); a larger set should prove more useful in determining the information necessary to allow the aid to function as hoped. Second, the experienced specialists used in the study had pretty much limited most of the search strategies before going online. Despite the disappointing experience in this study, the relevance feedback aid should prove to be of help to the casual user who ends up with a very broad search strategy as a result of using the first three aids. The relevance feedback procedure should then be of assistance by suggesting appropriate

\(^a\)Only 342 of the approximately 500 available profiles were used in this study because of file storage considerations (both internal and external) regarding the implementation of the profile-profile association aid. Additionally, upwards of 2000 profiles might be available which have been archived on tape.
strategies to narrow the search. Again, a study using the original evaluation design methodology should help to determine the true usefulness of this aid.

The most remarkable result from this research was the fact that, as shown in Table 7.6, the three association aids compositely generated potentially useful terms (or found a similar profile) for 78 (97%) of the profiles developed for the 80 randomly chosen MIC patrons in this study. This would seem to justify further research in this area to see how the aids described in this thesis could be further improved, or to see if additional aids could be developed.

Concerning the term-term association aids, several suggestions for future research come to mind. The first involves a study of the truncation problem discussed in Section 5.2. It would seem useful to examine this problem in depth from a conceptual point of view to determine how a truncated term should be treated in the association formulas, and to determine what effect various treatments might have on the results.

Another area worth looking into for the term-term (profile) association aid would be the effect of boolean logic on the associative relationships. (Only the unique terms of each profile were used when generating the term-term associations in this study). Since synonyms should occur in the same context in a profile (e.g., ORed with each other and/or ANDeD with a common term), this would seem to be a fruitful study to pursue. It is possible that the logic could be used to advantage to form smaller, but extremely relevant association lists.
A study of the effects of Boolean logic should also prove beneficial for the profile-profile association aid. In a vein similar to the truncation problem discussed in Section 5.2, if one profile contains only the word INFORMATION, while another contains only the pair INFORMATION AND RETRIEVAL, which profile is a subset of which? Again, a good conceptual study of this problem should prove to be very useful to future implementations of profile-profile association aids.

The unexpected success of the term-term aid based on documents suggests another area for future consideration. Since the presence of LC descriptors apparently contributed to the success of this aid, other data elements might be expected to contribute in a similar way. One possibility would be to include the words in the journal name, and also the journal codes, together with the title words and subject headings used in this research. Inclusion of these additional data elements might have an effect similar to that of the subject headings, since most journals are highly subject specific and words strongly associated with a particular journal would be expected to relate in some way to the subject area of the journal.

Finally, a few additional suggestions of a general nature should be mentioned. First, since this research used a test file (i.e., a sample set of search file documents) for interactive profile preparation, it would be interesting to study the effect of the test file size on the profile preparation process. In many cases the 5% sample seemed to be adequate, but often the specialist had to make an educated guess as to whether to broaden a search which retrieved few or no documents in the test file. Also, if the full retrospective
file could be used for evaluating the results of the coding process, a more accurate measure of the value of the Aids system might be obtained. In particular, it is felt that, in many cases where the Aids system provided useful terms which did not cause retrieval in the test file, useful documents might have been found had the full file been searched.

Second, a replication of the current research in other search environments with other databases would be of interest to see if similar results would be obtained. This would help further decide the utility of the profiling aids evaluated in this thesis.

Third, the evaluation of retrieval outputs based on the utility model discussed in Chapter 3 warrants further study. As discussed in Section 7.1.7, the graph developed in Section 3.4 worked well for modeling the decisions made during this study. If such graphs can be constructed for a large number of individual users, and then partitioned into groups of similar graphs which can be "averaged" to form "standardized" group graphs, such standardized graphs should prove most useful if important characteristics of a user belonging to any particular group can be isolated. If this can be accomplished, retrieval system evaluation could be simplified by limiting the extent to which user participation would be required for the evaluative process.

As a final concluding note it is hoped that the software developed in this research for online interactive querying of the MIC files will prove to be a useful by-product for the MIC information specialists regardless of which of the two systems - Aids or Standard - is used for profile development, when and if operational constraints permit.
Logon to System

The first step in using the on-line profiling system is to connect to the IRCC computer. To do this, dial 2-3950 and then type:

logon ts2066 size(256)  

where  represents depressing the "RETURN" key. The system will then prompt for the indicated information which should be entered as shown. (Note: In the following, computer output is shown in upper case and is also underlined. User input is shown in lower case.)

 PASSWORD? xxx  
 TERMINAL ID? 1m30  
 UNIVERSITY ID? yyyyyyy  

where xxx represents the three letter password and yyyyyyy represents the eight digit university id for MIC (these will be provided separately). After a few seconds, the computer will then respond

TS2066 LOGON IS PROGRESS AT time ON date  

After a few more seconds, the computer will further respond:

 READY  

At this point the profiling session can be started.

Please note that two sample sessions are provided at the end of the guide to further clarify any of the points discussed herein.
Initiating the Profiling Session

In response to the above "READY" prompt, the user should enter:

```
exec online  r
```

This invokes the profiling system which will respond:

```
WELCOME TO NIC-ONLINE - hh:mm:ss

LINE 1:
```

At this point the user may search the files or enter system commands as discussed below.

This first prompt may take up to 60 seconds as the computer loads the program and opens the appropriate files. After the initial prompt, the response should be fairly rapid.

Please note that the general form of the system prompt is

```
LINE xx:
```

where xx represents the current line number. This number will start with 1 and will be increased by 1 each time a new document set is defined (even if the set contains no documents). Should the user respond to any prompt with a system command rather than a search request, the line number will be reused for the following prompt.

A few special purpose prompts will be issued to request further information for some of the commands. These will be discussed in the appropriate section.

Searching the Files

The simplest type of search is the single term search. To employ this type of search, the user should simply enter the search term followed by (r). Following standard MIC conventions, the @ should precede an author's name, and a S should precede a journal coden. The system will respond with the number of articles containing the entered term followed by the retro adjustment (an estimate of the number of hits if the entire retrospective file is searched) in parentheses and finally the search term as entered. Any term longer than 12 characters will be automatically truncated for the file search, but the full term will show in the echo.
Note that there are several reserved words. These are the logical operators (e.g. AND) and the system command words (e.g. PRINT). Although most of these words are either stoplist words or too short to be used as search terms, should any of the reserved words be needed as search terms, simply enter the term in quotes.

Examples:  
**LINE 1:** psychology  
20 (  400) PSYCHOLOGY

**LINE 2:** "print"  
11 (  220) PRINT

**LINE 3:** print  
several document titles will be displayed here

Several search terms can be combined on the same line by using boolean operators. The operators should be inserted between the terms (infix notation), and each operator must be preceded and followed by a blank. The operators are:

AND logical AND (both must be present)  
OR logical OR (one or both must be present)  
NOT logical NOT (the first must be present, the second must not be present)

The above may be abbreviated by A, O, and N, respectively.

The default precedence is 1) AND  
2) OR  
3) NOT

so all AND's will be performed first followed by all OR's and finally all NOT's. Parentheses may be used to change the default order.

Eg. scale or test and intelligence  
    scale or (test and intelligence)  

to get (scale or test) and intelligence  

the user must insert the parentheses.
The system response to a boolean search will be similar to a single term only search, i.e. postings, retro adjustment, input line.

Eg. LINE 7: intelligence and test
6 ( 120) INTELLIGENCE AND TEST

In addition to the above, document sets already defined may be used in boolean combinations by using the line number corresponding to the desired document set.

Eg. LINE 7: test or scale
16 ( 320) TEST OR SCALE
LINE 8: intelligence and 7
8 ( 160) INTELLIGENCE AND 7

The set defined for line 8 will be the same as if intelligence and (test or scale) had been entered.

If several terms on the same line are to be combined with the same logical operator, a special shorthand notation may be used. The terms should be listed without the infix operators, and a "/op" should be entered at the end of the line, where op represents any of the three logical operators.

Eg. test scale measurement /o --> test or scale or measurement

Line Continuation

Should any file search or any system command require more than one line to enter, a "C" may be placed at the end of the first line preceded by a space. The system will respond with a C: prompt, and input may continue on the second line. Note, however, that the first line must end where the user would normally insert a blank - a single search term may not be split across lines. A maximum of two lines may be used for any single search term or command.
Displaying Documents at the Terminal

The PRINT command causes documents to be printed at the terminal. The format is:

\[ \text{print line\# format /items} \]

Notes: p may be used instead of print
line\# is the number of the line defining the set of documents to be printed. If line\# is omitted, the last document set defined will be used.
format codes are AU, TI, or ALL. If AU is specified, author name only will be listed. If TI is specified, title only will be listed. If AU, TI is specified, both author name and title are listed. If ALL is specified, author and title are listed along with journal information. If this field is omitted, the title only is printed.
items specify which documents in the set are to be printed. If the numbers are separated by commas, the indicated documents only are printed. Therefore, 1,3,6 will print the first, third, and sixth documents. A dash specifies a range of documents to be printed. 2-4 will print the second, third, and fourth documents. If this field is omitted, the first five documents will print (or all documents if the set contains fewer than five). Note that the slash (/) is required before this field.

Eg. print
\[ \text{print the first five titles from the most recent document set.} \]
print 7 /1
\[ \text{print the first title from the set defined on line 7.} \]
print 3 au,ti /1,6-10
\[ \text{print the first and sixth through tenth (inclusive) authors and titles from the set defined on line 3.} \]

Review Document Sets Already Created

By entering LIST or L, the user can display all lines through the current line. The computer will respond with:

\[ \text{LINE 1: line} \]
\[ \text{LINE 2: line} \]
\[ \text{etc.} \]
where line represents the line entered by the user. The postings and retro adjustment will not be displayed here.

**Ending the Session**

To end the session, simply hit the RETURN key in response to any LINE x: prompt. The computer will then reply:

**TO END HIT RETURN - TO CONTINUE TYPE "CONT"**

If the RETURN key was entered in error, type "CONT" and the computer will again respond with the line prompt and the session continues.

If RETURN is hit, the computer will leave the profiling system and return to TSO and give the READY prompt. If another profiling session is desired, simply enter EXEC ONLINE and a new profiling session will begin.

If the online use is complete, the user should then enter LOGOFF. The computer will respond with a two line message that the user has logged off. At this point, turn off the terminal and clear the phone line.

**Note:** If an unforeseen bug occurs during a profiling session and the computer does something strange, the READY prompt from TSO can be obtained by hitting the BREAK key twice. This should be used only in an "emergency" as the profiling session will stop immediately.

**MIC Related Features**

A few additional features exist, but are not discussed in this version of the manual. These features, such as an automatic conversion from the online profiling language format to the MIC profiling language format, are fairly specific to MIC and are not deemed to be of general interest.
SAMPLE SESSION #1

READY
exec online ← A

WELCOME TO MIC-ONLINE - 10:22:15

LINE 1: president
   1 ( 21) PRESIDENT

LINE 2: president*
   11 ( 231) PRESIDENT* ← E

LINE 3: carter or ford or nixon or kennedy
   10 ( 210) CARTER OR FORD OR NIXON OR KENNEDY

LINE 4: 2 and 3
   2 ( 42) 2 AND 3 ← G

LINE 5: print < ← H

PRESIDENTIAL POWER VERSUS BUREAUCRATIC INTRANSIGENCE - INFLUENCE OF THE NIXON ADMINISTRATION ON WELFARE POLICY

AMERICAN MOOD - 1979 - PRESIDENT CARTER 1ST 30 MONTHS PROVE A DISSAPPOINTMENT

LINE 5: list ← I

  LINE 1: PRESIDENT
  LINE 2: PRESIDENT*
  LINE 3: CARTER OR FORD OR NIXON OR KENNEDY
  LINE 4: 2 AND 3

LINE 5: ← J ← K

TO END HIT RETURN - TO CONTINUE TYPE "CONT"

LEAVING MIC-ONLINE
DON'T FORGET TO LOGOFF TSO - THANK YOU
Notes for Sample Session #1

A) TSO command to initiate online session.
B) Posting count (number of documents in file retrieved for this line).
C) Approximate number of documents in full retrospective file.
D) Echo of line.
E) Right truncation is available (the "*" is the truncation symbol).
F) The "or" can be abbreviated by "o".
G) Line numbers can be used in boolean combination.
H) Defaults are most recent line, title only, first five documents or all documents if fewer than five.
I) Review the session.
J) Since last line was a command (and not a search line), the line number is reused.
K) On this line the user entered the carriage return only.
L) Again, only the "return" key was depressed (this ends the session).
READY
exec online

WELCOME TO MIC-ONLINE - 14:27:59

LINE 1:psychology
    45 ( 945) PSYCHOLOGY

LINE 2:"print" ←----------------------A
    1 ( 21) "PRINT"

LINE 3:print ←----------------------B

VISUAL, VERBAL, AND SALES RESPONSES TO PRINT ADS

LINE 3:vice and president
    0 ( 0) VICE AND PRESIDENT←-----C

LINE 4:oil
    15 ( 315) OIL

LINE 5:p /3←------------------------D

WORLD-ECONOMY DEPENDENT ON OIL MONEY RECYCLING

LINE 5:p au,ti /3 ←----------------E

WORLD-ECONOMY DEPENDENT ON OIL MONEY RECYCLING
MURAMOTO S

LINE 5:p all /3←----------------F

WORLD-ECONOMY DEPENDENT ON OIL MONEY RECYCLING
MURAMOTO S
BUSINESS JAPAN
VOL. 24 NO. N10 , 1979 PAGE 35

LINE 5:intelligence a (test* o scale* o measure*) not stanford
    2 ( 42) INTELLIGENCE A (TEST* O SCALE* O MEASURE*) NOT STA
LINE 6: carter or c- G
C: kennedy
8 ( 168) CARTER OR KENNEDY H
LINE 7: carter kennedy nixon ford /or I
10 ( 210) CARTER KENNEDY NIXON FORD /OR
LINE 8:
TO END HIT RETURN - TO CONTINUE TYPE "CONT"
:
LEAVING MIC-ONLINE
DON'T FORGET TO LOGOFF TSO - THANK YOU

Notes for Sample Session #2

A) With the quotes included, the system will search the index using the term "print".
B) Without the quotes, this is the print command.
C) Zero posting also receives a line number.
D) Print the the third document from the most recent line (title only).
E) Print author and title for the third document from the most recent line.
F) Print all bibliographic information which is available for the third document from the most recent line.
G) This search line will continue onto the next line.
H) The system will concatenate the continued line.
I) The special end of line operator construction.
FILE STRUCTURES USED FOR THE AIDS PROGRAM

AND ALGORITHMS USED TO INVOKE THE AIDS
General Flowchart of Aids System Processing

Start

Read next input line

End of initial profile?

Another term?

Used before?

Add sums to prof-prof counts

Any associated terms?

Display associated term-lists

See the following algorithm for more detail.
1

Get selected line numbers

Any remaining lines?

Initiate Relevance Feedback Processing

Does user want more?

End

Modify search?

> 3 terms entered?

Any similar profiles?

Display one similar profile

2

3

noterms

ntere

yes

any

no

yes

no

no

yes

no
Algorithms used to Invoke the Aids

Algorithm for the three association aids:

1) Examine the user's next term.
2) If this term was examined earlier, go to Step 1.
3) Find Document Dictionary File page and swap in.
4) If term is in document dictionary, swap in Document Association File page and buffer associated term numbers.
5) Find Profile Dictionary File page and swap in.
6) If term is in profile dictionary, swap in Profile Association File page and buffer associated term numbers. Also note Profile Inverted File pointers. If term is not in profile dictionary go to Step 9.
7) Swap in Profile Inverted File page.
8) Add one to the count for each profile containing this term (the profile numbers are in the Profile Inverted File).
9) If the term was not in either dictionary, go to Step 1.
10) Display the association list(s). Note that this requires a look-up in the appropriate dictionary to translate the term numbers saved in Step 4 and/or Step 6 to the appropriate terms for display.
11) If fewer than 3 terms have been examined, go to Step 1.
12) If no old profile has association with the new profile above 0.7, go to Step 1.
13) Display the old profile with highest association to the new profile (if two or more are tied, display the profile with the largest number of unique terms).
14) Go to Step 1.

The files mentioned in the algorithm are described in Table B.1
Algorithm for the Relevance Feedback Aid (this is automatically invoked when the user enters a line with no terms to signal the end of the initial profile).

1) Prompt to determine which lines the user wishes to examine.
2) Find the line with the largest number of postings (if two or more are tied, select the first). If there are three or fewer postings on this line, stop.
3) Get the next article number from the internal buffer. (Go to 9 at end).
4) Display the title of the article.
5) Determine from user response whether the article is relevant, nonrelevant, or "don't know".
6) Find all terms used to represent this article from the Inverted Inverted File (accessed through the Relevance Feedback Pointer File).
7) Store these term numbers in the appropriate array based on the user's response from Step 5.
8) Go to Step 3.
9) Use the terms buffered in Step 7 to produce the display shown in Figure 5.3.
10) If user wishes to initiate processing for the next line, go to Step 2, else stop.
Table B.1

Files used by the Aids Program

<table>
<thead>
<tr>
<th>File</th>
<th>Field</th>
<th>Size (Bytes)</th>
<th>Description</th>
<th>Number of tracks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Profile Dictionary</td>
<td>Term</td>
<td>12</td>
<td>Contains one record for each unique term in the set of profiles.</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>Invptr</td>
<td>4</td>
<td>The alphabetic representation of the term.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Asnptr</td>
<td>4</td>
<td>A pointer to the Profile Inverted File. This points to the first record in that file for the term. The last record can be found by using the pointer for the next term.</td>
<td></td>
</tr>
<tr>
<td>Profile Inverted</td>
<td>Pronum</td>
<td>4</td>
<td>A pointer to the Profile Association File. Similar to the Invptr pointer.</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Contains a list of all profiles containing the term pointing to the list.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>The relative number of the profiles.</td>
<td></td>
</tr>
<tr>
<td>Profile Association</td>
<td>Tnum</td>
<td>4</td>
<td>Contains all associated terms and their measures for the term pointing to the list.</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>Meas</td>
<td>4</td>
<td>Relative number of the associated term.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Value of association between this term and the pointing term.</td>
<td></td>
</tr>
<tr>
<td>Document Dictionary</td>
<td>Term</td>
<td>12</td>
<td>Contains one record for each unique term in the set of documents.</td>
<td>94</td>
</tr>
<tr>
<td></td>
<td>Asnptr</td>
<td>4</td>
<td>The alphabetic representation of the term.</td>
<td></td>
</tr>
<tr>
<td>Document Association</td>
<td>Tnum</td>
<td>4</td>
<td>Contains all associated terms and their measure for the term pointing to the list.</td>
<td>76</td>
</tr>
<tr>
<td></td>
<td>Meas</td>
<td>4</td>
<td>Relative number of the associated term.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Value of association between this term and the pointing term.</td>
<td></td>
</tr>
</tbody>
</table>
Table B.1 (Continued)

<table>
<thead>
<tr>
<th>File</th>
<th>Field</th>
<th>Size (Bytes)</th>
<th>Description</th>
<th>Number of tracks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relevance</td>
<td>Docptr</td>
<td>4</td>
<td>Contains one record for each unique term in the set of documents.</td>
<td>9</td>
</tr>
<tr>
<td>Feedback</td>
<td></td>
<td></td>
<td>For the corresponding term (determined by relative location in the file), points to the set of documents containing the term. These documents are found in the Inverted Inverted File.</td>
<td></td>
</tr>
<tr>
<td>Pointer</td>
<td></td>
<td></td>
<td>xlabel</td>
<td></td>
</tr>
<tr>
<td>Inverted</td>
<td></td>
<td></td>
<td>Contains a list of all documents containing the term pointing to the list.</td>
<td>76</td>
</tr>
<tr>
<td>Inverted</td>
<td>Dnum</td>
<td>4</td>
<td>The relative number of the documents.</td>
<td></td>
</tr>
<tr>
<td>Profile</td>
<td></td>
<td></td>
<td>The profile file must also be referenced to obtain any profile which is found to be similar to the new profile by the Profile-Profile Association Aid. Since the profile format is unique to MIC, its structure will not be discussed here.</td>
<td>342\textsuperscript{a}</td>
</tr>
</tbody>
</table>

\textsuperscript{a} In MIC format, each profile requires one full track of disk space for storage. This is, of course, installation dependent.
APPENDIX C

SYSTEM DESCRIPTION OF THE CONSTRUCTION OF THE AIDS
Construction of the Files used by the
Profile-Profile Association and Term-Term (Profile) Aids

Each unique word in set of profiles followed by list of profiles which contain the word. (A relative number is assigned to each term).

All profile-term pairs (random order)

All profile-term pairs sorted by profile number.

Sparse profile-term matrix.

Array of single frequency counts

All term-term pairs (unsorted)
All term-term pairs sorted by first term, subordered by second term.

Sparse term-term co-occurrence matrix.

Sparse term-term association matrix.

All term-term pairs with measure (unsorted).

All term-term pairs with measure (sorted by term 1, then descending by value of measure).
These three files are online files used by the Aids system and are described in Appendix B.
Table C.1

Description of the Files used to Construct the Aids Based on the Profiles

<table>
<thead>
<tr>
<th>File</th>
<th>Field</th>
<th>Size (Bytes)</th>
<th>Description</th>
<th>Number of Records</th>
</tr>
</thead>
<tbody>
<tr>
<td>Profile</td>
<td></td>
<td>7290</td>
<td>A profile format unique to MIC; size of each profile is 7290 bytes.</td>
<td>342</td>
</tr>
<tr>
<td>Proword</td>
<td>Word</td>
<td>12</td>
<td>A word from the set of profiles.</td>
<td>5,055</td>
</tr>
<tr>
<td></td>
<td>Freq</td>
<td>4</td>
<td>Total number of times word occurs in the profile set.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Prof</td>
<td>2</td>
<td>Number of profiles in which word occurs.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Pname</td>
<td>*a 12</td>
<td>Profile name.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Rel</td>
<td>* 2</td>
<td>Relative profile number.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Cnt</td>
<td>* 2</td>
<td>Frequency of Word in profile.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Neg</td>
<td>* 2</td>
<td>Not used.</td>
<td></td>
</tr>
<tr>
<td>Insort</td>
<td>Prof</td>
<td>2</td>
<td>Relative profile number.</td>
<td>9,058</td>
</tr>
<tr>
<td></td>
<td>Term</td>
<td>4</td>
<td>Relative term number.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Fill</td>
<td>12</td>
<td>Not used.</td>
<td></td>
</tr>
<tr>
<td>Outsort</td>
<td></td>
<td>same as Insort</td>
<td>Relative profile number.</td>
<td>9,058</td>
</tr>
<tr>
<td>Tape</td>
<td>Prof</td>
<td>2</td>
<td>Relative profile number.</td>
<td>342</td>
</tr>
<tr>
<td></td>
<td>Nterms</td>
<td>2</td>
<td>Number of terms which follow.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Tnum</td>
<td>* 4</td>
<td>Relative term number.</td>
<td></td>
</tr>
<tr>
<td>Sing</td>
<td>Counts</td>
<td>2</td>
<td>An array of 5,055 single frequency counts.</td>
<td>5,055</td>
</tr>
<tr>
<td>Prtape</td>
<td>Term1</td>
<td>4</td>
<td>First term of pair (relative no.).</td>
<td>252,195</td>
</tr>
<tr>
<td></td>
<td>Term2</td>
<td>4</td>
<td>Second term of pair (relative no.).</td>
<td></td>
</tr>
<tr>
<td>Sorpr</td>
<td></td>
<td>same as Prtape</td>
<td>Relative term number.</td>
<td>252,195</td>
</tr>
<tr>
<td>Cotape</td>
<td>Tnum</td>
<td>4</td>
<td>Relative term number.</td>
<td>243,453</td>
</tr>
<tr>
<td></td>
<td>Count</td>
<td>2</td>
<td>Set to -1 if this is a first term, else the co-occurrence count of this term with first term.</td>
<td></td>
</tr>
<tr>
<td>Tanimot</td>
<td>Tnum</td>
<td>4</td>
<td>Relative term number.</td>
<td>14,897b</td>
</tr>
<tr>
<td></td>
<td>Meas</td>
<td>4</td>
<td>Set to -2 if this is a first term, else the measure of association between this term and the first term.</td>
<td></td>
</tr>
</tbody>
</table>
Table C.1 (Continued)

<table>
<thead>
<tr>
<th>File</th>
<th>Field</th>
<th>Size (Bytes)</th>
<th>Description</th>
<th>Number of Records</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sortrec</td>
<td>Term1</td>
<td>4</td>
<td>First term (relative number).</td>
<td>19,876</td>
</tr>
<tr>
<td></td>
<td>Term2</td>
<td>4</td>
<td>Second term (relative number).</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Meas</td>
<td>4</td>
<td>Association measure between the two terms.</td>
<td></td>
</tr>
<tr>
<td>Sortout</td>
<td></td>
<td></td>
<td></td>
<td>19,876c</td>
</tr>
</tbody>
</table>

a Indicates a repeating field.

b After terms which co-occur only once have been deleted.

c The CPIDICT program removes all pairs whose measure of association is less than 0.250. For the set of profiles used in the study, this filter reduced the number of term-term pairs from 19,876 to 9,508.
Construction of the Files used by the Term-Term (Document) Aid

MIC Inverted File

DICTSING
- Dict
- Insort

SORT1
- Outsort

MAKEART
- Mafie

TPRSING
- Sing
- Prtape

An alphabetical array of all unique terms in the set of documents. The position in the array assigns a relative number to each term.

All term-document pairs (random order)

All term-document pairs sorted by document number.

Each document number followed by all terms used to describe the document.

Array of single frequency counts.

All term-term pairs (unsorted).
All term-term pairs sorted by first term, subordered by second term.

Sparse term-term co-occurrence matrix.

Sparse term-term association matrix.

All term-term pairs with measure (unsorted).

All term-term pairs with measure (sorted by term 1, then descending by value of measure).
These two files are online files used by the Aids system and are described in Appendix B.
<table>
<thead>
<tr>
<th>File</th>
<th>Field</th>
<th>Size (Bytes)</th>
<th>Description</th>
<th>Number of Records</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inverted</td>
<td>-</td>
<td>-</td>
<td>The inverted file format used by MIC.</td>
<td>73,861</td>
</tr>
<tr>
<td>Dict</td>
<td>Words</td>
<td>12</td>
<td>An array of the 73,861 unique terms found in the documents.</td>
<td>73,861</td>
</tr>
<tr>
<td>Insort</td>
<td>Term</td>
<td>4</td>
<td>Relative term number.</td>
<td>243,329</td>
</tr>
<tr>
<td></td>
<td>Art</td>
<td>4</td>
<td>Relative document number.</td>
<td></td>
</tr>
<tr>
<td>Outsort</td>
<td>same as Insort</td>
<td></td>
<td></td>
<td>243,329</td>
</tr>
<tr>
<td>Mfile</td>
<td>Art</td>
<td>4</td>
<td>Relative article number.</td>
<td>24,878</td>
</tr>
<tr>
<td></td>
<td>Nterms</td>
<td>2</td>
<td>Number of terms which follow.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Tnum</td>
<td>*a</td>
<td>Relative term number.</td>
<td></td>
</tr>
<tr>
<td>Sing</td>
<td>Counts</td>
<td>2</td>
<td>An array of 73,861 single frequency counts.</td>
<td>73,861</td>
</tr>
<tr>
<td>Prtape</td>
<td>Term1</td>
<td>4</td>
<td>First term of pair (rel. no.).</td>
<td>1,390,476</td>
</tr>
<tr>
<td></td>
<td>Term2</td>
<td>4</td>
<td>Second term of pair (rel. no.).</td>
<td></td>
</tr>
<tr>
<td>Sorted</td>
<td>same as Prtape</td>
<td></td>
<td></td>
<td>1,390,476</td>
</tr>
<tr>
<td>Cofile</td>
<td>Word</td>
<td>4</td>
<td>Relative term number</td>
<td>1,116,251</td>
</tr>
<tr>
<td></td>
<td>Count</td>
<td>2</td>
<td>Set to -1 if this is a first term, else the co-occurrence count of this term</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>with the first term.</td>
<td></td>
</tr>
<tr>
<td>Tanimot</td>
<td>Tnum</td>
<td>4</td>
<td>Relative term number.</td>
<td>177,133</td>
</tr>
<tr>
<td></td>
<td>Meas</td>
<td>4</td>
<td>Set to -2 if this is a first term, else the measure of association between</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>this term and the first term.</td>
<td></td>
</tr>
<tr>
<td>File</td>
<td>Field</td>
<td>Size (Bytes)</td>
<td>Description</td>
<td>Number of Records</td>
</tr>
<tr>
<td>--------</td>
<td>-------</td>
<td>--------------</td>
<td>-------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>Sortfl</td>
<td>Term1</td>
<td>4</td>
<td>First term (relative number).</td>
<td>214,510</td>
</tr>
<tr>
<td></td>
<td>Term2</td>
<td>4</td>
<td>Second term (relative number).</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Meas</td>
<td>4</td>
<td>Association measure between the two terms.</td>
<td></td>
</tr>
<tr>
<td>Sortout</td>
<td></td>
<td></td>
<td></td>
<td>214,510&lt;sup&gt;c&lt;/sup&gt;</td>
</tr>
</tbody>
</table>

<sup>a</sup> Indicates a repeating field.

<sup>b</sup> After terms which co-occur only once have been deleted.

<sup>c</sup> The CDICT program removes all pairs whose measure of association is less than 0.010. For the set of documents used in this study, this filter reduced the number of term-term pairs from 214,510 to 119,278.
Construction of the Files used by the Relevance Feedback Aid

See the Term-Term. (Document) Construction flow chart and Table C.2 for details.

These two files are online files used by the Aids system and are described in Appendix B.
APPENDIX D

USE OF MIC WEIGHTING FORMULA IN A PREDICTABLE MANNER

D.1 General Formula for the Weighting Function

\[
T SV = f(w_1, w_2, \ldots, w_n) = \frac{w_1 w_2 \cdots w_n}{w_1 w_2 \cdots w_n + (1-w_1)(1-w_2)\cdots(1-w_n)} \quad (D.1)
\]

where \(w_1, w_2, \ldots, w_n\) are arbitrary weights assigned to profile terms (or groups of terms) found in the title or index set of any document.

D.2 Derivative of a Generalized Approximation from Some Specific General Cases

Single Term Match with Term Weight \(w_1\)

\[
f(w_1) = \frac{w_1}{w_1 + (1-w_1)} = w_1 \quad (D.2)
\]

Two Term Match, One of Whose Weights is 0.5 (Neutral Weight)

\[
f(w_1, 0.5) = \frac{w_1(0.5)}{w_1(0.5) + (1-w_1)(1-0.5)} = \frac{w_1(0.5)}{(0.5)(w_1 + (1-w_1))} = w_1 \quad (D.3)
\]

Note that the neutral weight has no effect on the value of the function regardless of how many neutral weight terms are found.

Two Term Match, One of Whose Weights is the Absolute Negative Weight 0.0

\[
f(w_1, 0.0) = \frac{w_1(0.0)}{w_1(0.0) + (1-w_1)(1-0.0)} = 0.0 \quad (D.4)
\]

\[\text{Based on the technique proposed by Petrarca (36).}\]
Two Term Match with Term Weights $w_1$ and $w_2$

$$f(w_1, w_2) = \frac{w_1 w_2}{w_1 w_2 + (1-w_1)(1-w_2)}$$  \hspace{1cm} (D.5)

Table D.1

<table>
<thead>
<tr>
<th>$w_1$</th>
<th>$w_2$</th>
<th>$f(w_1, w_2)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.480</td>
<td>0.480</td>
<td>0.46064 = (0.460)</td>
</tr>
<tr>
<td>0.490</td>
<td>0.490</td>
<td>0.48008 = (0.480)</td>
</tr>
<tr>
<td>0.500</td>
<td>0.500</td>
<td>0.50000 = (0.500)</td>
</tr>
<tr>
<td>0.510</td>
<td>0.510</td>
<td>0.51992 = (0.520)</td>
</tr>
<tr>
<td>0.520</td>
<td>0.520</td>
<td>0.53993 = (0.540)</td>
</tr>
<tr>
<td>0.530</td>
<td>0.530</td>
<td>0.55997 = (0.560)</td>
</tr>
<tr>
<td>0.540</td>
<td>0.540</td>
<td>0.57949 = (0.579)</td>
</tr>
<tr>
<td>\vdots</td>
<td>\vdots</td>
<td>\vdots</td>
</tr>
<tr>
<td>0.590</td>
<td>0.590</td>
<td>0.67435 = (0.674)</td>
</tr>
<tr>
<td>0.510</td>
<td>0.520</td>
<td>0.52997 = (0.530)</td>
</tr>
<tr>
<td>0.510</td>
<td>0.490</td>
<td>0.50000 = (0.500)</td>
</tr>
<tr>
<td>0.520</td>
<td>0.490</td>
<td>0.51000 = (0.510)</td>
</tr>
</tbody>
</table>

Note from Table D.1 that as long as $w_1$ and $w_2$ have values close to the neutral weight of 0.5, the weighting function for the two term case may be simplified to the Approximation Formula D.6 shown below. The greater the difference between the $w$'s and 0.5 the greater the deviation from the above approximation. Note also that positive weights ($>0.5$) are neutralized by negative weights ($<0.5$) of equal magnitude.

$$f(w_1, w_2) \approx w_1 + w_2 - 0.500$$  \hspace{1cm} (D.6)

Three Term Match with Term Weights $w_1$, $w_2$, and $w_3$

$$f(w_1, w_2, w_3) = \frac{w_1 w_2 w_3}{w_1 w_2 w_3 + (1-w_1)(1-w_2)(1-w_3)}$$  \hspace{1cm} (D.7)
Table D.2

Table of Function Values for Some Specific Values of $w_1$, $w_2$, and $w_3$ for a Three Term Match

<table>
<thead>
<tr>
<th>$w_1$ = $w_2$ = $w_3$</th>
<th>$f(w_1, w_2, w_3)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.510</td>
<td>0.529968 = (0.530)</td>
</tr>
<tr>
<td>0.520</td>
<td>0.559745 = (0.560)</td>
</tr>
<tr>
<td>0.530</td>
<td>0.589145 = (0.589)</td>
</tr>
</tbody>
</table>

From Table D.2 it can be seen that the three term case, as before, may be simplified to the following approximation.

$$f(w_1, w_2, w_3) \approx w_1 + w_2 + w_3 - 2(0.500) \quad (D.8)$$

Generalized Approximation Formula for the MIC Weight Function

$$f(w_1, w_2, ..., w_n) \approx w_1 + w_2 + ... + w_n - (n-1)(0.500) \quad (D.9)$$

D.3 Illustrations using the Generalized Approximation Formula D.9

Several examples based on those in Table 4.1 (see Section 4.1) will be used to illustrate the use of the generalized approximation formula D.9. As seen in Table D.3, example 1, a document represented by the term CARS, but not by the term CHRYSLER, would receive a weight of 0.510 and would be retrieved. A document discussing CHRYSLER and CARS would receive a weight of 0.500 ($0.510 + 0.490 - 0.500$) and would not be retrieved. However, a document discussing CHRYSLER CARS and LIGHT TRUCKS would receive a weight of 0.510 ($0.510 + 0.510 + 0.490 - 1.00$) and would be retrieved. Thus, a strategy which exactly duplicates the boolean form of the request can be constructed.

As seen in Table D.3, example 2, a document represented by the terms DATA and STRUCTURE would receive a weight of 0.530 ($0.510 + 0.520 - 0.500$) and would be retrieved. However, a document discussing a FILE of DATA would receive a weight of only 0.520 and would not be retrieved. Also, a document discussing the STRUCTURE of an ORGANIZATION would receive a weight of 0.520 ($0.520 + 0.520 + 0.480 - 1.00$) and would not be retrieved. On the other hand, a document describing FILE STRUCTURE and ORGANIZATION would receive a weight of 0.530 ($0.510 + 0.520 + 0.520 + 0.480 - 1.50$) and would be retrieved. Similar computations can be performed for the other terms in the profile to show that the two strategies are equivalent.
Table D.3

Two Examples of Coding a Profile using the Generalized Approximation Formula

<table>
<thead>
<tr>
<th>Example</th>
<th>Boolean Query</th>
<th>Thresh- Line</th>
<th>Query Term 1</th>
<th>Query Term 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(CARS NOT CHRYSLER) OR (LIGHT AND TRUCKS)</td>
<td>.510</td>
<td>.510 CARS</td>
<td>.510 LIGHT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.490</td>
<td>.490 CARS</td>
<td>.510 CHRYSLER</td>
</tr>
<tr>
<td>2</td>
<td>(DATA OR INFORMATION OR FILE) AND (STRUCTURE OR ORGANIZATION OR MANAGEMENT)</td>
<td>.530</td>
<td>.510 DATA</td>
<td>.510 INFORMATION</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.510</td>
<td>.510 FILE</td>
<td>.520 STRUCTURE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.520</td>
<td>.520 ORGANIZATION</td>
<td>.520 MANAGEMENT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.480</td>
<td>.480 STRUCTURE</td>
<td>.480 STRUCTURE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.480</td>
<td>.480 ORGANIZATION</td>
<td>.480 MANAGEMENT</td>
</tr>
</tbody>
</table>
BIBLIOGRAPHY

1. Bayer, Bernard, Manger, Mechanized Information Center and Former Director of TIDB, Personal Communication.


161


22. Lazorick, Gerald, Former Manager of MIC and Former Director of TIDB, Personal Communication.


GLOSSARY

Notation

Di : Represents a document in the corpus.
n : Number of retrieved documents in a search.
P : The precision of a search (r/n).
Qi : Represents a profile.
r : The number of relevant retrieved documents in a search.
\bar{r} : The number of nonrelevant retrieved documents in a search.
R : The recall of a search.
Ri : Represents the retrieval vector in associative retrieval.

SA : The Asymmetric Subset similarity measure

SD : Doszkocs' similarity measure.

ST : The Tanimoto similarity measure.

SV : The vector product similarity measure

\text{t} : The total number of terms in a set of documents or a set of profiles.

Ti : Represents a term used to represent a document, or a term contained in a profile.

v_{ij} : Is 1 if term j is contained in profile i; 0 otherwise.

w_{ij} : Is 1 if term j is used to index document i, 0 otherwise.
Definitions

Document Retrieval System: A retrieval system capable of retrieving bibliographic data in response to search requests.


Profile: Used as a synonym for Search Request.

Query: Used as a synonym for Search Request.

Search Request: A statement of the user's question coded in a form required by a computerized retrieval system.

User Interest Profile: Used as a synonym for Search Request.