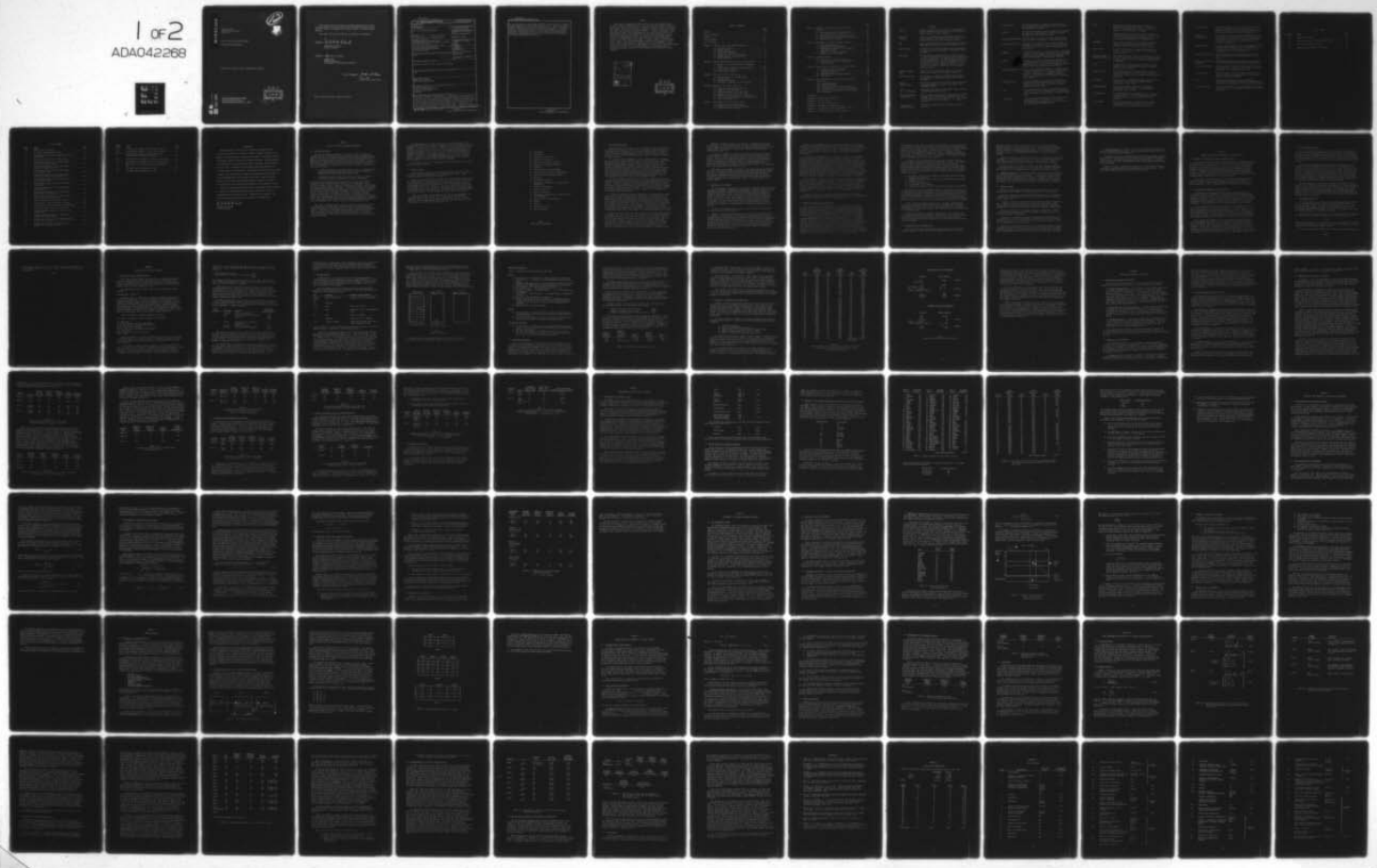
AD-A042 268 OHIO STATE UNIV COLUMBUS DEPT OF COMPUTER AND INFORM--ETC F/G 5/2
CIRC II DATA BASE CLASSIFICATION.(U)
JUN 77 L J WHITE, A E PETRARCA, L G CRAWFORD F30602-76-C-0102
UNCLASSIFIED RADC-TR-77-211 NL

1 OF 2
ADA042268



ADA042268



RADC-TR-77-211
Final Technical Report
June 1977

CIRC II DATA BASE CLASSIFICATION

The Ohio State University

Approved for public release; distribution unlimited.

DDC
RECEIVED
JUL 26 1977
D

ROME AIR DEVELOPMENT CENTER
Air Force Systems Command
Griffiss Air Force Base, New York 13441

AD No.
DDC FILE COPY

This report has been reviewed by the RADC Information Office (OI) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

This report has been reviewed and is approved for publication.

APPROVED: Nicholas M. Difondi

NICHOLAS M. DIFONDI
Project Engineer

APPROVED: Howard Davis

HOWARD DAVIS
Technical Director
Intelligence & Reconnaissance Division

FOR THE COMMANDER: John P. Huss

JOHN P. HUSS
Acting Chief, Plans Office

Do not return this copy. Retain or destroy.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: RADC-TR-77-211	2. GOVT ACCESSION NO. / 3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): CIRC II DATA BASE CLASSIFICATION	5. TYPE OF REPORT & PERIOD COVERED: Final Technical Report, 1 Jan 76 - 31 Mar 77
	6. PERFORMING ORG. REPORT NUMBER: N/A
7. AUTHOR(s): Dr. Lee J. White, Barry J. Brinkman, Dr. Anthony E. Petrarca, Sanjay Mittal, Laurel G. Crawford	8. CONTRACT OR GRANT NUMBER(s): F30602-76-C-0102
9. PERFORMING ORGANIZATION NAME AND ADDRESS: The Ohio State University/Department of Computer and Information Science, 2036 Neil Avenue Mall, Columbus OH 43210	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 62702F, 31025F, 45940114
11. CONTROLLING OFFICE NAME AND ADDRESS: Rome Air Development Center (IRDT), Griffiss AFB NY 13441	12. REPORT DATE: June 1977
	13. NUMBER OF PAGES: 101
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office): Same	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE: N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Same

18. SUPPLEMENTARY NOTES

RADC Project Engineer:
Nicholas M. DiFondi (IRDT)

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Automatic Classification
Subject Category Classification
Document Classification
Sequential

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report describes the development of a classification system for the CIRC II Data Base. 98 CIRC II classes are designed which partition the documents of this data base. The software which assigns these classes to incoming documents utilizes a sequential classification algorithm. In this approach, only as much of each document is read to accurately assign one or more classes, together with a confidence probability for each assigned class. In this way, a compromise is obtained between efficiency and accuracy. A number of parameters are available in this software to effect this trade off. Additional software → next page

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

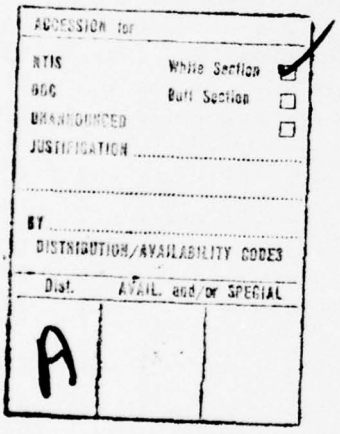
407 121

cont

has been developed to analyze sample documents to define the CIRC II classes, producing keywords and frequency distributions over the classes. This software provides flexibility for the classification system, as a class can be added or deleted, a class modified by submitting additional documents, or the keyword selection criterion can be altered. A number of experiments were conducted using this classification system on CIRC II documents. It was shown that satisfactory classification could be achieved, and a stable set of keywords and frequency distributions obtained.

PREFACE

This report constitutes the final report of work accomplished under Air Force Contract F30602-76-C-0102. The contractor would like to thank the technical monitor of this project, Mr. Nicholas Difondi, of Rome Air Development Center. He has been helpful on numerous occasions, and has contributed many ideas for resolving procedural problems. A number of personnel at FTD have worked closely in the development of this classification system. Mr. John McElroy and Mr. William Mace are from computer operations, and were helpful in indicating how a number of computer systems problems could be solved. Mr. Donald Quigley was a primary contact at FTD, and helped to formulate the objectives of this study. He made a number of contributions in the course of the system development, especially during Phase I of this study. Special appreciation is expressed to Ms. Helen Thompson, who was the principal contact during the implementation of Phase II of this development. She was very cooperative in the many requests for information and data required during this contract, and spent extra hours in response to these requests.



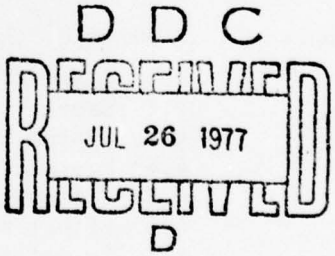


TABLE OF CONTENTS

		PAGE
Preface		i
Table of Contents		ii
Glossary		iv
List of Figures		viii
List of Tables		ix
Chapter 1	CIRC II Activity and Problem Definition	1
1.1	The CIRC II System	1
1.2	COSATI Classes	2
1.3	The Problem Statement	4
1.4	Approach to the Problem	5
1.5	Summary of Results and Conclusions	6
1.6	Recommendations for Future Work	7
1.7	Report Overview	8
Chapter 2	General Description of Sequential Classification	10
2.1	Results of the National Science Foundation Study	10
2.2	An Overview of Sequential Classification	10
2.3	Bayes Distance Software	11
2.4	Classification Output	11
Chapter 3	Selection of the 110 UDC Classes	13
3.1	Description of the 110 UDC Classes	13
3.2	The UDC Parser	15
3.3	UDC Sample Documents	17
3.4	Selection of Keywords for the UDC Study	19
Chapter 4	Results of the Phase I UDC Study	23
4.1	Classification Evaluation Criteria	23
4.2	Sample vs. Test Documents	23
4.3	System Parameters for the UDC Study	24
4.4	Evaluation of the Phase I UDC Study	25
4.5	Effects of System Parameters on Results	28
4.6	Effect of Dropping Classes from Consideration	29
4.7	Selecting Classes Early	30
Chapter 5	Development of the 98 CIRC II Classes	32
5.1	Expansion of the COSATI Classes	32
5.2	Manual Selection of Sample Documents	33
5.3	Analysis of Sample Documents	34

PAGE

Chapter 6 CLASSIFY - The Sequential Classification Algorithm 39

6.1 The Sequential Classification Approach 39
6.2 Input Information for CLASSIFY 39
6.3 The Sequential Classification Algorithm 41
6.4 Confidence Levels and Termination Criteria 43
6.5 Modifications to CLASSIFY 44

Chapter 7 KEYFINDER - The Sample Documents Analyzer 47

7.1 The KEYFINDER Software 47
7.2 Required Input for KEYFINDER 48
7.3 SORT/MERGE and PHASE3 - The Sorting and Counting Functions 48
7.4 CONVERT - Selection of Keywords 52
7.5 Modifications of KEYFINDER 52

Chapter 8 Compound Keywords 55

8.1 Definition of Compound Keywords 55
8.2 Construction and Use of the Compound Keyword Tables 56

Chapter 9 Classification of Documents of Varying Subjects 60

9.1 Documents Which Change Subject 60
9.2 Bayesian Distance Classifier 60
9.3 Compound Documents 62
9.4 Experiments with Compound Documents 63
9.5 Conclusions 64

Chapter 10 Final Experiments with the CIRC II Classes and Conclusions 65

10.1 Final Experiments 65
10.2 Keyword Selection 65
10.3 Evaluation of CIRC II Classification 68
10.4 Timing Measures for CIRC II Classification 72
10.5 BAL Version of CLASSIFY for IPIR Documents 73
10.6 Conclusions 74

References 76

Appendix A COSATI Class Frequencies A1

Appendix B 110 UDC Classes B1

Appendix C UDC Class Frequency Data C1

Appendix D A Sampling of Keywords by Class D1

Appendix E A Sampling of Compound Keywords by Class E1

Appendix F Final Definition of CIRC II Classes F1

GLOSSARY

Alpha (α)	An input threshold to the sequential algorithm which is used to determine which classes are to be retained for classification of a given document.
Alpha Test (α-test)	A test of the sequential algorithm which utilizes the alpha parameter to determine which classes are to be retained for classification of a given document.
BAL	Basic assembly language for the IBM 360/65; the CLASSIFY algorithm was to be coded in both BAL and PL/I form.
Bayes Distance	A classification criterion used in this investigation; it could be potentially applied to Intelligence Report documents.
Bayes Rule	This statistical technique allows the calculation of a posteriori probability given the relevant measurement a priori probabilities. In this situation, given class a priori probabilities, and observed keywords within the document and their a priori probabilities of being in a specific class C_j, Bayes rule allows the calculation of the updated probability of this document being in class C_j.
CIRC II Classes (98)	These classes are the final product of this study, and are to be applied to the CIRC II Data Base.
CIRC II Data Base	The data base for which the classification system is to be applied, containing scientific and technical documents.
CIRC II Output Format	The off-line printed output from the CIRC II Data Base is in this format; it is used as the input to the KEYFINDER software.
Class (C_j), or Category	A grouping of documents in the CIRC II Data Base which contain similar subject matter.
Class a Posteriori Probability (α_j)	After a number of keywords have been read from a given document, this is the updated probability that the document should be placed into class C_j. This concept is synonymous with the confidence level of class C_j.
Class a Priori Probability (q_j)	The initial probability that a document belongs in class C_j.

Classification	The partitioning of a document data base into sets called classes, where each class consists of documents of similar subject content.
CLASSIFY	The software developed in this contract which implements the sequential algorithm; this classifies incoming documents into the 98 CIRC II classes.
Compound Keyword (CKW)	A keyword phrase consisting of two or three adjacent words; it will be treated as a single keyword concept.
Confidence Level (α_j)	A measure of the confidence that a CIRC II Class C_j assigned to a document is correct; this is synonymous with the concept of the a posteriori probability α_j for class C_j.
CONVERT	A software systems which is considered part of KEYFINDER; it takes the word frequency table obtained by KEYFINDER using the sample documents, and selects keywords to be used in the sequential classification algorithm.
COSATI	Committee on Scientific and Technical Information of the Federal Council on Science and Technology; the COSATI codes are 22 numeric codes designating specific areas of scientific and technical information.
Default Probability (δ)	A very small probability assigned to a keyword as an a priori probability $P(W_i \mid C_j)$ when no sample document in class C_j contained keyword W_i; this is required because a zero probability would not allow correct operation of the sequential classification algorithm.
Frequency Table	The table produced by the software KEYFINDER when analyzing the sample documents, and consists of the number of times each word occurred in the sample documents for each class.
FTD	Foreign Technology Division, an organization within the Air Force which is responsible for the administration, processing, and development of the CIRC II Data Base.
Hash Table	A data structure which allows a very rapid search for information about a word detected in an input document; because of this speed, it is used in both the CLASSIFY and KEYFINDER software.

IPIR	Input Processor Intercommunication Record; the machine record in which each CIRC II element is formatted for input processing, and will be the input format for the BAL version of CLASSIFY.
IR	Intelligence Report; a specific report originated or disseminated by intelligence collection agencies; it is apt to change subject often within the report, and this causes special problems for classification.
KEYFINDER	The software developed in this study which analyzes sample documents, producing keywords and their frequency distributions over the CIRC II Classes.
Keyword (W_i)	A word or phrase used for document classification because it is indicative of the class to which that document belongs.
Keyword a Priori Probability $P(W_i \mid C_j)$	The probability that a particular keyword W_i will occur given a document from class C_j; this is obtained using frequency count data from the frequency table produced by KEYFINDER.
PL/I	A high-level programming language available on IBM computers; all the developed software is originally written in this language.
Primary Class	One or more classes to which the subject content of a document pertains in a major way; this concept was used in the evaluation of the sequential classification algorithm.
R Parameter	This parameter of KEYFINDER determines the number of keywords to be read from the document between applications of the α-test.
Sample Documents	The CIRC II documents submitted to KEYFINDER for analysis to produce keywords and their frequency distribution over classes.
Scaling Factor	A factor applied in the computation of the α-test to avoid underflow, i.e., the generation of very small probabilities which cannot be stored within the computer.
SDI Profiles	Selective Dissemination of Information user input, which allows the CIRC II documents in the user's interest areas to be brought to his attention.

Secondary Class	One or more classes which are relevant to the subject content of document, not in a major way, but in a peripheral sense; this concept was used in the evaluation of the sequential classification algorithm.
Sequential Classification	A classification method which was implemented for the CIRC II Data Base in this study; it is called sequential because only a portion of each document is read before a classification decision is made, resulting in a savings in computer time.
Sequential Test	A test of the sequential classification algorithm which determines which classes are to be retained for classification of a given document; this concept is synonymous with that of the alpha-test.
T--Document Threshold	The initial number of keywords which must be read from a document before the first sequential test is applied.
Technical Information Specialist	The personnel who assist the CIRC II user in preparing and using SDI profiles and retrospective searches.
Termination Criteria	For the sequential algorithm, the termination criteria determines when a classification decision can be made, and no more of the document need be read.
Test Documents	CIRC II documents which were used to evaluate the CLASSIFY software, but which were not also analyzed as sample documents by KEYFINDER.
UDC	Universal Decimal Code; a numeric method of subjectively categorizing information, assigned by the originator and predominant in open literature documents.
UDC Classes (110)	The classes developed in the first phase of this study, primarily aimed at classifying documents from the open literature.

LIST OF FIGURES

FIGURE	TITLE	PAGE
3.1	Tables for UDC Parser	16
7.1	Frequency Table Where Count Has Overflowed	50
8.1	Three Compound Keyword Tables	56
8.2	Compound Keyword Tables - An Example	58

LIST OF TABLES

TABLE	TITLE	PAGE
1.1	COSATI Subject Groupings	3
3.1	UDC Sample Document Analysis Data	18
3.2	Distribution of Words Over Classes for the 110 UDC Classes.	20
3.3	Keyword Criteria for the 110 UDC Classes	21
4.1	Classification Results for the UDC Classes Using 1,837 Sample Documents	26
4.2	Classification Results for the UDC Classes Using 9,563 Sample Documents	26
4.3	Classification Results for the UDC Classes as Evaluated by FTD	27
4.4	Classification Results for the UDC Classes Varying the Final Selection Criterion	28
4.5	Classification Results for the UDC Classes for Two Keyword Sets	28
4.6	Classification Results for the UDC Classes for Varying δ	29
4.7	Classification Results for the UDC Classes With Parameter T	29
4.8	Classification Results for the UDC Classes When Classes are Retained	30
4.9	Classification Results for the UDC Classes When Classes are Selected Early	31
5.1	Sample Documents for the CIRC II Classes	35
5.2	Distribution of Words Over CIRC II Classes	36
6.1	Termination Criteria Experiments with UDC Classes	45
7.1	Those Words Where Class Counts Overflowed	49
9.1	Compound Document Experiment - Selection of a Primary Class	63
9.2	Compound Document Experiment - Required to Satisfy the Criteria Twice	64
10.1	Keyword Set Selection Without Compound Keywords for the CIRC II Classes	66
10.2	Keyword Set Selection Including Compound Keywords for the CIRC II Classes	67

TABLE	TITLE	PAGE
10.3	Classification Results for the CIRC II Classes	70
10.4	Classification Timings for the CIRC II Classes	73
10.5	Classification Results for BAL CLASSIFY on IPIR Documents for CIRC II Classes	74
C1	Distribution of Documents by UDC at One Digit Root	C2
C2	Distribution of Documents by UDC at Two Digit Root	C3
C3	Distribution of Documents by UDC at Three Digit Root	C5
C4	Four Digit Root Distribution for UDC	C7
C5	Five Digit Root Distribution for UDC	C7

EVALUATION

The application of The Ohio State University automatic subject classification software to a sample of CIRC II documents resulted in the design of a desirable subject classification scheme. The design consisted of 98 subject classes of which 87 classes were defined. The establishment of a sound subject classification for CIRC II data base characterization will aid in the elimination of shortcomings experienced with the use of present subject classifications. The subject classification can be used to qualify user profiles when documents are disseminated via the CIRC II Selective Dissemination of Information system and to qualify retrieval requests via the CIRC On-Line document retrieval system. The degree of accuracy that this classification scheme will lend remains to be seen.

When the opportunity arises, FTD plans to experiment with the software to finalize the classification scheme and to evaluate its effect on dissemination and retrieval accuracy. This work is in support of the written word exploitation mission as defined in TPO/Thrust R3D.

Nicholas M. Difondi

NICHOLAS M. DIFONDI
Project Engineer

CHAPTER 1

CIRC II ACTIVITY AND PROBLEM DEFINITION

1.1 The CIRC II System

The Central Information Reference and Control (CIRC) II System is a document reference system in the area of natural sciences and engineering. Responsibility for this system is charged to the Foreign Technology Division (FTD), an organization within the Air Force which is responsible for the administration, processing, and development of the CIRC II System. The CIRC II data base now references in excess of four million documents; its growth rate is approximately 25,000 references each month. Access to the data base is through two modes:

a. a current awareness service which apprises users of documents newly acquired during regular update periods by means of a Selective Dissemination of Information (SDI) profile, and

b. retrospective searches which are made either on-line or off-line to locate documents corresponding to specific requirements.

Both the profile system and the on-line system can attain very specific information such as personalities, facilities or nomenclature. Also, selection criteria, such as country of publication or document type or date, are used to obtain explicit references. An important aspect of the matching of a document with a profile or a search is the use of a document classification in order to better identify those groups of documents which are most likely to yield specific documents of interest to the user, and avoid superfluous retrieval. The most complex aspect of document matching, however, involves concepts, composed of words, groups of words, or qualified words. More complicated concepts can be constructed using Boolean connectives. Users of the CIRC II system are assisted in these many aspects of retrieval by an FTD representative, called a Technical Information Specialist.

The types of documents that constitute the CIRC II data base and are continually input into the CIRC II data base vary from journal articles taken from available foreign journals to technical reports and intelligence reports. These types of documents normally assume various formats and CIRC II provides an input processor which converts these to a standard IPIR (Input Processor Intercommunication Record Format) for various CIRC II system processing.

A classification of a document data base is the partitioning of the documents into sets called classes or categories, where each class consists of documents of similar subject content. There are presently two classifications utilized in the CIRC II Data Base: the COSATI and UDC classification codes. The COSATI classification was produced by the Committee on Scientific and Technical Information of the Federal Council on Science and Technology, and consists of 22 numeric codes designating specific areas of scientific and technical information. The Universal Decimal Codes, or UDC, is a numeric method of subjectively categorizing information assigned by the originator, and is predominant in open foreign literature documents.

In the next section, the relationship of the COSATI codes in the CIRC II System is further explored.

1.2 COSATI Classes

As new incoming documents are processed and become part of the CIRC II Data Base, they are assigned one or more COSATI subject codes. These classification codes are indicated in Table 1.1.

There are two problems with the COSATI classification which necessitated the development of a new classification system. First, it can be seen that the classes are too broad; they need to be subdivided in order to allow a more specific indication of subject area. The second problem is that four of the COSATI codes, viz., 05, 07, 13, and 20, account for over 50% of all the documents in the open literature. This clearly indicates that documents are not distributed at all evenly over the COSATI codes. More details about distribution statistics of the COSATI codes are provided in Appendix A.

These two problems are addressed specifically in the development of the final CIRC II test classes. These classes may be considered as subdivided COSATI codes, and indeed are organized in that way. However, experience with the UDC classification was also taken into account in the final design of the CIRC II test classification.

01	Aeronautics
02	Agriculture
03	Astronomy and Astrophysics
04	Atmospheric Sciences
05	Behavioral and Social Sciences
06	Biological and Medical Sciences
07	Chemistry
08	Earth Sciences and Oceanography
09	Electronics and Electrical Engineering
10	Energy Conversion (Non-Propulsive)
11	Materials
12	Mathematical Sciences
13	Mechanical, Industrial, Civil and Marine Engineering
14	Methods and Equipment
15	Military Sciences
16	Missile Technology
17	Navigations, Communications, Detection, and Countermeasures
18	Nuclear Science and Technology
19	Ordnance
20	Physics
21	Propulsion and Fuels
22	Space Technology

TABLE 1.1
COSATI Subject Groupings

1.3 The Problem Statement

The objective of this contract was to establish a subject classification of the CIRC II Data Base in order to aid in document dissemination by the CIRC II Selective Dissemination of Information (SDI) System. In addition, a classification processing system was to be developed based upon these classes in order to assign appropriate subject areas to documents entering the CIRC II System.

The reason for the development of such a classification is that many search terms have several meanings, and their specific meaning is generally dependent upon the context of the document in which they occur. This ambiguity, when document context is a factor, can be removed by the use of CIRC II qualifiers or the Boolean connective "AND". However, an improved classification could in one way be used to resolve this ambiguity, since search for the documents relevant to a user's interest area could concentrate within specified classes. Within each class, terms would be much less ambiguous and the terms employed could be more specific to the user's interest. In addition, the classes themselves could be used as an appropriately modified search key for document retrieval.

The newly developed classification would have to remedy the problems observed in the COSATI classes in Section 1.2. An increased number of classes would be required to provide the greater specificity missing in the COSATI classes. The CIRC II documents should be distributed more evenly over the new classes, providing both better document retrieval and efficient computer processing in finding relevant documents. This could eventually be applied to the entire CIRC II Data Base in order to simplify the processing of retrospective and other searches.

In order to accomplish either the objectives of developing a new comprehensive classification for the CIRC II Data Base or a processing system to assign these classes to incoming CIRC II documents, it was necessary to demonstrate that such objectives were indeed feasible. It had to be shown that a classification could be defined that would satisfactorily partition the documents from the CIRC II Data Base. It further had to be demonstrated that, using this classification, the documents of the data base could automatically be placed into the correct classes. It was required that this classification be accomplished as accurately as possible, and yet the process be efficient in the sense of the number of documents processed per unit time.

Notice that these objectives imply a number of trade offs which needed to be studied. The size of the classification needs to be increased over that of the COSATI classes. The larger is the subject classification, then the finer the partitioning of the data base which could be achieved. However, as the number of classes increase, and the classes become increasingly specific, then a given document may have to be assigned to many of these very specific classes, and many documents may be too general to fit into such specific classes. Also the cost will increase with the number of classes. A balance needs to be achieved in the number of classes selected.

Similarly a tradeoff exists in the software to automatically classify the documents. It should classify as accurately as possible, and yet the analysis should not be so extensive that it requires excessive time to process a given number of documents. In this study a method was applied which addressed this trade off.

The original statement of work for this project called for the subject classification to exhibit near uniform distribution of documents across subject categories, and a significant number of categories should be represented. This uniform distribution requirement later had to be modified to allow certain subjects of greatest interest to CIRC II System users to be given more emphasis and visibility as distinct classes somewhat out of proportion to the number of documents in these subject areas. This compromise had to be addressed by FTD and this contractor.

With regard to the number of categories, FTD system constraints indicated that about one hundred classes would yield the desired degree of specificity, and yet not too many more than one hundred classes should be selected, or else serious processing problems would be encountered. Thus part of the feasibility study would be to ascertain whether satisfactory classification could be obtained with an appropriately chosen set of approximately one hundred classes.

1.4 Approach to the Problem

The classification structure was to be derived using Ohio State University's computerized sequential classification algorithm. A modified version of this algorithm would then be produced to process incoming CIRC II documents by assigning appropriate classes. This algorithm was developed through the support of the National Science Foundation, the final reports of which are given as references [7,11]. The reason the algorithm is called sequential is that only as much of the input document is read to accurately classify the document, thus obtaining a most accurate classification decision as efficiently as possible. This addresses the accuracy-efficiency trade off indentified in Section 1.3, and is why this algorithm was proposed for this problem.

In order to accomplish the objectives of this contract, the one-year effort was divided into two phases:

PHASE I: A research and development phase was first conducted to show the feasibility of the approach. An initial classification structure was constructed, and keyword selection techniques and the sequential classification algorithm applied to CIRC II documents provided by FTD. This was a six month effort, which demonstrated that satisfactory classification of the CIRC II documents could be achieved. It was clear that subject classification had to be modified, primarily to contain those classes for subject areas of most interest to CIRC II users, even though the relative number of documents in those areas might be quite small.

PHASE II: An implementation phase was planned, during which time the software was modified and recoded so as to be operational on the FTD IBM 370/65 computer. The software was to be installed and tested at the FTD facility. In addition, the subject classification was considerably modified, keyword selection improved, and an approach to analyze Intelligence Reports was investigated.

In terms of the methodology of the approach of Phase I, this required that FTD provide this contractor with a frequency distribution of CIRC II documents over the Universal Decimal Classification (UDC) System for a large typical group of documents. FTD also provided ten to twenty thousand typical document abstracts in computer-readable form which had already been assigned UDC codes. This allowed the design of a UDC-based subject classification, and the abstracts provided the required sample documents. Using these sample abstracts, keywords were extracted which characterized each class. A number of experiments were performed using the sequential classification algorithm, and these experiments demonstrated the feasibility of the development of a satisfactory set of CIRC II classes. The sequential classification algorithm was shown to constitute a feasible approach for classifying CIRC II documents.

The initiation of Phase II was based upon the satisfactory classification of CIRC II documents achieved in Phase I. However, the subject classification has to be considerably modified to expand the number of classes for subject areas of greatest interest to CIRC II users. The sample documents were carefully selected from various sources using CIRC II search procedures, and not just from the open foreign literature documents. These documents were screened manually to ensure that each class was characterized as well as possible.

On March 31, 1977, the modified software and tapes of the defining documents, word frequencies, keyword and keyword frequencies, were all turned over to FTD for possible utilization at their facility.

1.5 Summary of Results and Conclusions

This study has shown the feasibility of the development of about one hundred classes which will satisfactorily classify the CIRC II Data Base. The CIRC II classes developed in this study represent a compromise between the requirement that the documents be approximately uniformly distributed over the classes and the requirement that other specific areas of greater interest to the users of the CIRC II System be chosen as distinct classes.

A classification system has been developed which will analyze CIRC II documents and assign them accurately and efficiently to one or more CIRC II classes. It has been shown that more than 80% of the assigned classes are correctly assigned and yet over 20 documents per second could be automatically classified on the FTD IBM 360/65 computer. (Only correct assignment of test documents to classes is reported). It was found that no more than five CIRC II classes was required to be assigned to any document, and so the output of this classification system consists of one to five classes and their confidence levels. The confidence level of a class corresponds to a confidence probability that the class is correct after a sufficient portion of

that document has been read. For nearly all documents, it was quite typical that confidence levels exceeding 0.85 were achieved before ten keywords were read within the document. It was found that the highest accuracy was obtained when the class with the highest confidence level was chosen. However, in order that more appropriate classes be chosen, a compromise was made in this regard. Experiments also showed that the best performance was obtained when classes were chosen at the end of the analysis of each document, rather than selecting classes earlier in the analysis.

A software system was developed and delivered to FTD which analyzes sample documents, and thus defines each of the CIRC II classes. The primary purpose of this software is to produce keywords and their frequency distributions over the classes which are used in the classification system. Data was delivered which included all documents analyzed so that FTD could reproduce the results of this study, or modify those results. This software allows the overall classification system to be dynamic, and the following changes can be made:

1) new CIRC II classes can be defined by submitting additional sample documents for analysis;
2) additional documents can be submitted to further define an already existing class;
3) a class can be deleted;
4) the keywords can be changed;
5) a number of parameters of the classification system can be changed.

It was found that about 4,000 keywords were sufficient to satisfactorily classify nearly all CIRC II documents. This is a level which does not exert an excessive demand for storage when the classification system is operated in a production environment. At this level, nearly all documents with text contain a sufficient number of keywords so that a satisfactory classification decision can be made. Furthermore, it was found that this number of keywords was fairly stable, and did not require modification with small system changes.

It was found that for most classes, about 150 sample documents were sufficient to adequately define that class for classification. Stability was achieved with this figure or less, and this requires that only about 15,000 sample documents be analyzed for 100 classes, which is not unreasonable for a one-time project.

A method was developed, though not implemented in this software, for classifying documents which frequently change subject. Intelligence Reports will often tend to have this property, and thus unmodified sequential classification may only obtain a partial view of the subject areas discussed in such a document.

1.6 Recommendations for Future Work

One of the most serious problems during this study is that defining sample documents for each class had to be manually selected or at least

manually screened. This is very tedious, and yet good defining sample documents for each class are essential. It is recommended that a method be found to automatically generate such documents. One technique which might prove useful here is the Bayes distance criterion briefly described in Chapter 9, as it is a very sensitive indicator of the subject content of a document.

Another area where more research is needed is in better keyword selection techniques. It was found in this study that certain high frequency words had to be manually removed from the keyword set. Repeated efforts failed to remove these words by any automatic method, i.e., algorithmic technique.

The work begun in this study on applying the Bayes distance criterion to documents which change subject should be completed. An evaluation should then be made to see if implementation is required for the CIRC II System.

Compound keywords are two or three adjacent words chosen to provide more specific discrimination for classification. A facility for compound keywords was included in the classification software. A systematic study should be made for the CIRC II System to see if the extra complexity of this facility is justified in terms of substantially improved classification. There was insufficient time during this contract period to perform such an evaluation.

1.7 Report Overview

Chapter 1 has provided an introduction to the CIRC II classification problem and how this problem was approached during this contract study.

Chapter 2 indicates a general description and overview of the sequential classification algorithm which resulted from a National Science Foundation research grant [7,11].

A report of the work on Phase I of this contract is given in Chapters 3 and 4. Chapter 3 indicates how the UDC classes were selected and used, while Chapter 4 reports the experiments from Phase I and their evaluation. Appendices B and C give the UDC classes and their frequencies, respectively.

The work of Phase II is contained in Chapters 5 through 10. Chapter 5 describes the development of the final 98 CIRC II Classes, and they are presented in detail in Appendix F.

Chapter 6 is a detailed discussion of the delivered software version of the sequential classification algorithm. An indication of its performance characteristics are given, along with what sort of modifications can be made both in parameters and in the software.

Chapter 7 is detailed discussion of the software which analyzes sample documents and produces keywords and frequency distributions which characterize the CIRC II classes. This is important in that additional sample documents may be submitted by FTD which were not previously available for this study.

A compound keyword is a phrase of two or three adjacent words which are useful for document classification. The compound keyword software developed in this study is discussed in Chapter 8.

Intelligence Reports are documents which potentially may change subject matter several times within the text and this may pose problems for sequential classification. Although not implemented in the final system, an approach will be documented to deal with documents which change subjects several times within their text.

Chapter 10 reports experimental results with the final CIRC II classes and keywords, and indicates the system performance characteristics. Conclusions and recommendations for future work are presented.

CHAPTER 2

GENERAL DESCRIPTION OF SEQUENTIAL CLASSIFICATION

2.1 *Results of the National Science Foundation Study*

The author has been involved in a two-year study of a sequential analysis model for automatic document classification [7,11], which was supported by the National Science Foundation. It is based on the notions of sequential analysis as described by Wald [10], concepts which have been successfully applied in the field of pattern recognition (see, for example, the work of Fu [6]). Fried and his co-workers originally suggested this approach be applied to automatic classification in 1968 [5]. An initial implementation of this idea was constructed by Aberi [1], but the recent research effort has provided numerous improvements to this implementation.

As a result of this NSF study, the classification algorithm was available in PL/I form, and thus could be applied to the CIRC II data base as soon as a number of input format problems were resolved. This was the reason that Phase I could be completed in as short a period as six months.

2.2 An Overview of Sequential Classification

The sequential classification method assumes the availability of a given number of subject categories, a selection of keywords representative of these categories, and the a priori probabilities of all keywords representative of these categories, and the a priori probabilities of all keywords within each category. Given the categories, these probabilities are usually determined from a representative sample set of documents by counting the frequencies of all keywords within the sample documents of each category.

In the sequential approach, only as much of each document to be classified is read until it can be classified into one or more categories. A word in the document is isolated, and compared to a list of keywords. If it is not a keyword, another word is isolated and read. If it is a keyword, then access is made to a frequency table to obtain its a priori probability within each category. As this is done repeatedly with successive keywords, an a posteriori probability is calculated for each class using Bayes rule.

The a posteriori probability for each remaining class is compared to some predetermined threshold. If it is less than this threshold, then the class is dropped. When a termination condition is achieved, all classes with sufficiently high a posteriori probabilities are assigned to that document. If the end of the document is encountered before the termination condition is achieved, the document is deemed unclassifiable. Details of the termination condition and how "sufficiently high" a posteriori probabilities should be for a class to be selected will be discussed in Chapter 6.

2.3 Bayes Distance Software

One of the most interesting results of the NSF study is a method by which to identify "noisy" keywords during the sequential classification of a document. Since the sequential approach means that only a few keywords are read before the document is classified, this process may be susceptible to several keywords near the beginning which are either incorrect indicators of the content of that document or else indicate a different aspect of the document.

In Chapter 9, the Bayes distance measure is formally defined. In reference [7] its properties are explored, where it is shown that this measure is related to probability of error at each stage and also to the uncertainty or entropy, but the main point here is that the Bayes distance measure is extremely sensitive to an unexpected keyword. For example, in an ideal document in which all initial keywords are indicative of single class and the sequential algorithm arrives rapidly at a correct decision, this measure is found to rise rapidly and monotonically to a unity value. For a document containing a keyword inconsistent with that class, there is an immediate observable drop in the Bayes distance measure.

The Bayes distance classification algorithm is based upon this idea, and throws out keywords which have been identified by the Bayes distance as "noisy". More is given on this algorithm in Chapter 9, and in reference [7]. It has been used in parallel with the sequential algorithm several times during this work, but is primarily suggested for use with Intelligence Reports (IR), where the subject matter may change several times within the document, and it is felt that the sensitive Bayes distance criterion should be able to detect this change.

2.4 Classification Output

As each document is analyzed by the sequential algorithm, new records will be created and added to the end of each document, one new record for each assigned class. This information will consist of at most five classes and their confidence levels. These confidence levels are the a posteriori probabilities discussed in Section 2.2, and sum to unity over all classes remaining in contention at any time.

The format of each class entry is a five character field; for example, if class 49 is selected with confidence .86, the output entry will be

04986.

Note that three digits are allocated for the class code, since the software allows for expansion of the current 98 classes. The confidence level can be expressed to two significant decimal figures.

In case the document is deemed unclassifiable, the output will appear as

00000.

In the special case that there is no abstract text and the document is unclassifiable based on the title (or there is also no title), the output will appear as

99900.

CHAPTER 3

SELECTION OF THE 110 UDC CLASSES

3.1 Description of the 110 UDC Classes

The Universal Decimal Classification (UDC) is a complete hierarchical numerical classification, and is described in detail in reference [9]. Since it is utilized by much of the rest of the world for classification of technical material, many of the CIRC II documents from the open literature already have one or more assigned UDC codes.

The hierarchical nature of these codes can be illustrated by several examples.

Ball bearings: 621.822.7

The first digit, 6, indicates this subject is within applied science, medicine, and technology. The first two digits, 62, narrow this down to engineering and technology. The first three digits, 621, narrow it still further to mechanical and electrical engineering. For readability, the UDC convention is to insert a period every three digits. Thus 621.8 denotes power transmission and materials handling. 621.82 indicates transmission systems and parts. 621.822 denotes bearings and bushings, and finally code 621.822.7 specifically identifies the subject of ball bearings. This clearly shows the hierarchical nature of the decimal classification code.

The second example is briefly explained as follows:

Pest Control of wheat by chemical spraying: 632.934:633.11

63 Agriculture
632 Plant Diseases and Pests, Crop Damage
632.9 Pest Control, Plant and Crop Protection
632.93 Pest Control Measures
632.934 Pest Control by Chemicals (Spraying)
633 Field Crops
633.1 Cereals, Corn, and Grain Crops
633.11 Wheat

Note that modifiers can often be handled as distinct UDC codes, treated as different aspects of the subject matter. Note that the colon (:) serves as an articulation point between UDC codes, although plus (+) can also be used in this way.

It was because of this hierarchical nature of the UDC codes and that many of the open literature CIRC II documents were already assigned such codes that the initial set of classes were selected on the basis of the UDC schedule. This contractor was provided by FTD the distribution statistics of 208,815 CIRC II documents by UDC code. This distribution is given in

Appendix C. First the document distribution by UDC at one digit root is shown; this clearly illustrates that 96.8% of these documents are in codes 5 and 6:

5	Math and Natural Science	31.1%
6	Applied Science, Medicine, and Technology	65.7%
		96.8%

The document distribution is then indicated by two, three, and four digit roots where such breakdowns are required to specify distributions to less than 1% of the total number of documents.

These distributions could then be used in order to choose about one hundred classes so as to satisfy the specification that no class should contain more than 1% of the documents. Of course, some care had to be taken to put UDC codes of similar subject matter together in one class, and to avoid artificial boundaries between classes as much as possible, both in order to find a small number of characterizing keywords which could accurately classify documents into that class.

Another consideration in the construction of the 110 UDC classes was to include a number of general classes. This was done to allow the classes to be truly hierarchical, and also is indicated in the UDC frequency distribution given in Appendix C. Several examples are:

UDC CLASS NO.	UDC CODE	SUBJECT	% DOCUMENT DISTRIBUTION
9	53 only	Physics and Mechanics	0.16
	530	General Principles of Physics	0.14
	531	Mechanics	0.96
			1.26
47	6 only	Applied Science, Medicine, Technology	----
	60 only	General Technology	0.05
	62 only	Engineering and Technology	1.42
			1.47

This decision, however, was not a good idea, because classification of documents into such classes was unacceptable to FTD. There will have to be some class into which such documents are placed, but the problem was that too many documents with specific subject content were being placed in these general classes. The new CIRC II classes described in Chapter 5 were designed to eliminate this problem.

The 110 UDC classes are listed and described in Appendix B. As can be seen, the largest class contained 1.47% of the documents, whereas the smallest corresponded to 0.32%, thus satisfying the requirement of near uniform distribution of documents across subject categories. There is a substantial emphasis on UDC codes in the range 5 and 6 as discussed before. In most cases, a UDC class consists of consecutive UDC codes, but this is not

universally true. In some cases a more homogeneous class can be constructed from nonconsecutive UDC codes, and also a better delineation obtained between classes. In the next section a description of the UDC parser is given, which constitutes a mapping between the UDC codes and these 110 UDC classes.

3.2 The UDC Parser

The importance of the UDC parser is that sample documents with an assigned UDC code can be automatically chosen as sample documents. In the case of the CIRC II sample documents described in Chapter 5, they had to be selected manually because of the absence of a hierarchical code like the UDC code assigned to each document.

Before discussing the way the UDC parser works, further detailed features of the UDC codes need to be identified. The main UDC code may be modified by the following qualifiers:

CODE	INDICATES	PURPOSE OF QUALIFICATION
.01/.09 -0/-9	Special Analyticals	qualifies the UDC code further
.00	Viewpoint	
(1/9)	place	indicates location
" "	time	indicates a date or chronological time
(0)	form	format of document
=	language	indicates race or language
'1/9'	synthetic numbers	concatenates several UDC codes to form a compound concept

Recall the major articulation points between distinct codes are : and +; as indicated above, / indicates an inclusive continuation.

In general, the UDC parser works by extracting a single UDC code and ignoring all the modifying information above. The special format of these modifiers makes the separation possible, i.e., look for .05, .005, (439), " ", (089), =, and ' '. One further complication is that several UDC codes may be grouped together with parentheses, and this must be differentiated from the place and form modifiers. This is handled in the algorithm indicated below. Another difficulty which had to be dealt with was UDC keypunching errors. The major design consideration here was to be able to recover from such errors, going on to the next UDC code if several are assigned. If no UDC code is obtained, the document is skipped. In this case, some UDC codes may be lost. Our experience with this UDC parser is

less than 0.1% of the UDC codes were lost due to keypunching errors, and although a number of simplifications were made in the algorithm, there was no instance where a UDC code was parsed incorrectly.

After the UDC code is extracted from any modifiers, a data structure as illustrated in Figure 3.1 is used. The first UDC code digit identifies the entry in the first table. Except for '5' or '6' as this digit, this first table entry contains the UDC class information. If this digit is a '5' or '6', a pointer to another table allows the examination of the second decimal digit of the code. Pointers to other tables are followed until a specific class is identified as indicated by the UDC classes in Appendix B. In all, twenty tables are required for these 110 UDC classes. The eleventh entry in each table is used for the "only" classes discussed in the previous section.

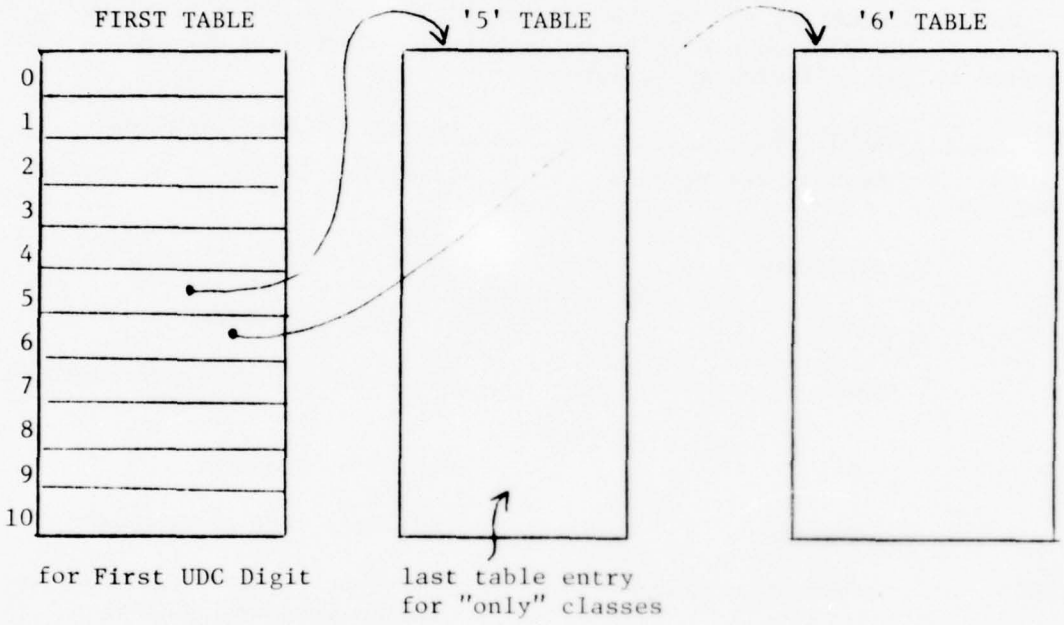


FIGURE 3.1
Tables for UDC Parser

The following is a simplified summary of the UDC parser algorithm. Notice that it is a two-pass parser, examining each UDC code twice.

UDC Parser Algorithm

Input: A character string representing a UDC code.

Pass 1:

1) Any initial left parenthesis is ignored (grouping is assumed).
2) Each digit or . is moved to the current buffer except as noted below.
3) Anything within parentheses is ignored (except initial left parentheses, as we assume all others are modifiers of the UDC code).
4) All characters within double quotes are ignored.
5) All numerals and . are ignored after =, -, /, or '.
6) There is a change to a new buffer after : or + (a new UDC code is assumed).
7) All numerical characters and . are ignored after any alphabetic character.
8) All stray right parentheses are ignored.
9) If any other character except a blank is encountered, an error message is written, an error code returned, and the parsing is terminated (the UDC code is in error, and the document will be skipped).
10) Stop processing at the first blank.

Pass 2:

11) For each buffer constructed in Pass 1, read the decimal digits, ignoring periods, and utilize the twenty tables to find the appropriate UDC class.
12) If no class is defined at the end of the buffer, then the correct UDC class is the "only" 11th entry in the current table.

Termination and Output

13) The UDC classes are then sorted and like classes combined, since a number of UDC codes assigned to the document may all correspond to the same single class.
14) The UDC class numbers are returned in an array with a separate variable either indicating the number of classes returned or an error code. At most five classes are returned.

3.3 UDC Sample Documents

There are two approaches to the selection of sample documents. One is to carefully select documents to represent each class, allowing specific control of how many sample documents are chosen for each class. The other approach is to analyze a large number of documents which are sufficiently representative of the data base as a whole, and utilize the UDC parser to associate each document with the proper class or classes. There are disadvantages to each approach. The advantage of the first approach is that you

can be sure that each class is well represented in terms of sample documents, but the problem is that by carefully selecting the documents in this way, the documents may only represent restricted aspects of the subject class. The random approach will have this advantage of representing all aspects of the subject class, as long as the documents one works with are sufficiently representative of the entire data base. With the random approach, however, some classes may not have enought documents to represent it.

The random approach was utilized in constructing the sample documents for the UDC classes. Later during the project it was realized that it was impossible to construct a representative set of documents which would satisfy all users, and thus the sample documents for the CIRC II classes are constructed using the class-by-class approach described in Chapter 5.

Both of these basic problems could be alleviated if a very large number of documents could be analyzed. But in Chapter 7, the analysis software is described, and it can be seen that it would be very costly to analyze hundreds of thousands of documents, for example, when this is not necessary to well define the classes. Furthermore, there are overflow problems if word counts exceed certain levels.

Initially a sample of 22,491 open literature CIRC II documents were made available by FTD. The following was ascertained:

Number of Documents with UDC Codes:	10,936
Number of Documents with no Text:	12,904
Number of Documents with UDC Codes and Text:	9,563

Thus a file of 9,563 potential sample documents was established. Note that a surprisingly high percentage of the documents have no text or assigned UDC codes. Initially 1,837 of these documents were analyzed to get some initial data in order to evaluate parameters for the analysis software described in Chapter 7. Table 3.1 shows this data compared to the entire set of sample documents, the balance of which were analyzed later. A token is defined to be the occurrence of a word in a sample document. Thus this is considerably larger than the number of distinct words.

NUMBER OF DOCUMENTS ANALYZED	NUMBER OF DISTINCT UDC CODES ANALYZED	TOKENS OBTAINED	DISTINCT WORDS OBTAINED	STOPLIST SIZE
1,837	2,424	99,000	12,066	3,200
9,563	(not available)	712,710	34,058	902

TABLE 3.1 UDC Sample Document Analysis Data

First note that in the analysis of the 1,837 documents, a stop list of 3,200 words was used. Since it was found that only 1,930 of the stoplist words even occurred in these documents, a reduced stoplist of 902 words was utilized for the remainder of the sample documents.

As mentioned above, the documents in this randomly chosen sample were not distributed evenly over the 110 UDC classes. This is illustrated in Table 3.2, which indicates the distribution of words over those classes for which it occurred in at least one sample document. These may be considered as potential keywords for that class. Each word will in general occur in more than one class, as can be seen that these words co-occur in the classes a total of 203,218, whereas there are only 34,058 distinct words.

Class 106 has 5,234 words, whereas no documents were found at all for class 49. This disparity lead to some classification problems, but in Chapter 4, it will be seen that reasonably good classification results were obtained despite this difficulty.

3.4 Selection of Keywords for the UDC Study

A number of authors, most recently Salton [8], have shown that the best keywords are those of intermediate frequency words. Thus, words of high frequency and low frequency should be eliminated. High frequency words occur in documents from all classes, tend to have low discriminant power, and are quite likely to be ambiguous. Low frequency words do not occur often enough to be useful, even though intuitively they might seem quite indicative of one class or several classes, since too much storage would be required for these words.

It will be convenient to define the following parameters describing the frequency distribution of a word over the 110 UDC classes within the sample documents:

F	Total Word Frequency
D	Number of Documents Word Occurred In
CT	Total Number of Classes with Nonzero Frequency Count
C1	Number of Classes with Unit Frequency Count
C2	Number of Classes with Frequency Count of Two

In terms of these parameters, Table 3.3 shows a number of keyword criteria which were applied to either the 12,066 or 34,058 word sets. A quantitative evaluation of most of the keyword sets for classification will be given in Chapter 4, but our objective here is to compare keyword selection criteria. A qualitative evaluation of classification effectiveness will suffice to indicate that aspect of the comparison.

For the 3,585 keyword set in Table 3.3, the first criterion is a low frequency threshold, and is clearly effective, eliminating 8,400 of the initial 12,066 words. The second criterion is the high frequency cutoff, and is not effective at all, eliminating only 81 words. This motivates a better

UDC CLASS	WORDS INDICATING THAT CLASS	UDC CLASS	WORDS INDICATING THAT CLASS	UDC CLASS	WORDS INDICATING THAT CLASS
1	1241	38	581	75	1851
2	722	39	927	76	2985
3	1330	40	1590	77	1727
4	1127	41	917	78	2330
5	974	42	2668	79	1520
6	1487	43	2650	80	1916
7	812	44	1206	81	1490
8	679	45	278	82	3454
9	3140	46	440	83	1701
10	1689	47	3124	84	425
11	1017	48	3536	85	2161
12	1567	49	0	86	1365
13	1980	50	2117	87	1076
14	1806	51	1974	88	1563
15	1045	52	2100	89	1856
16	1208	53	3618	90	3659
17	1892	54	648	91	2139
18	1157	55	3027	92	1330
19	427	56	1846	93	1848
20	1523	57	2045	94	763
21	2240	58	2344	95	2840
22	232	59	986	96	1644
23	1296	60	2325	97	1198
24	2638	61	1244	98	2935
25	3001	62	2072	99	3329
26	3714	63	1477	100	428
27	3193	64	1890	101	721
28	1761	65	2812	102	2150
29	2103	66	3083	103	1727
30	1078	67	2831	104	4176
31	785	68	3389	105	1192
32	458	69	2714	106	5234
33	664	70	4679	107	419
34	1111	71	1956	108	3121
35	1418	72	4331	109	1745
36	2177	73	2029	110	373
37	1307	74	1374	TOTAL OVER ALL CLASSES	203218

TABLE 3.2
Distribution of Words Over Classes for Which it Occurred in at Least One Sample Document from that Class, for the 110 UDC Classes

APPLIED TO THE 12,066 WORD SET

CRITERIA	WORDS REMOVED	
$D \geq 3$	8400	3585 KW
$CT-C1-C2 \leq 15$	81	
$D \geq 3$	8400	
and either $F \geq 20$		
or $F/CT \geq 2.5$	347	3319 KW
or $(F-C1-2C2)/(CT-C1-C2) \geq 4.5$		
$D \geq 3$	8400	
$CT > C1+C2$	1329	1990 KW
$F \geq 20$	347	

APPLIED TO THE 34,058 WORD SET

CRITERIA	WORDS REMOVED	
$F \geq 20$	29,548	
$D \geq 8$	35	
$CT > C1+C2$	59	3333 KW
either $F/CT \geq 2.5$		
or $(F-C1-2C2)/(CT-C1-C2) \geq 4.5$	1,083	
$F \geq 33$	30,749	3309 KW
$CT < 70$	299	

TABLE 3.3
Keyword Criteria for the 110 UDC Classes

high frequency cutoff in the 3319 keyword set, whereas a frequency (F) - oriented low frequency cutoff is used. The 3319 keyword set does represent an improvement, and gave the best classification results of all the keyword sets based upon the 12,066 word set. A final 1990 keyword set examined another set of criteria; note especially that the CT>C1+C2 criteria eliminates a substantial number of words. The problem is that too many words have been eliminated and the 1990 keyword set gives inferior classification results as will be indicated in the next chapter. This criteria will be successfully applied in the 34,058 word set, however.

For the 34,058 word set, the above keyword extraction experience was applied to obtain the 3333 and 3309 keyword sets. Notice that although substantially different criteria were used, nearly identical keyword sets were obtained. This illustrates some degree of stability in keyword selection. The 3333 keyword set was somewhat superior in classification performance, and in the next chapter is considered essentially the final keyword set for the UDC classes. The 3309 keyword set has the virtue, however, of possessing a very simple criteria set.

Careful examination of the high frequency criteria for all the keyword sets indicates a problem in automatically rejecting high frequency words which are not keywords, and yet retaining a sufficient set of keywords to classify most documents. All the keyword sets contain words which are intuitively known not to be good keywords, but no automatic keyword selection method seems to be able to differentiate them from more desirable high frequency words. The most effective method is to place these words on the stoplist in the first place, and this was done in the processing of sample documents for CIRC II classes described in Chapter 5.

CHAPTER 4

RESULTS OF THE PHASE I UDC STUDY

4.1 Classification Evaluation Criteria

In order to assess the performance of the sequential classification method for the 110-class UDC study, the criteria used were as follows:

a) UDC Matching Criterion - This criterion can be obtained objectively, and counts a document classified correctly if at least one of the classes chosen by the sequential classifier matches a UDC code which had been assigned to the document. This particular criterion is rather severe and restrictive since many of the classes chosen by the sequential classifier may actually be correct even though they had not been assigned to the document by the author or indexer. Consequently two additional criteria were used based on a subjective analysis of whether the classes chosen by the sequential classifier were correct or not. These criteria are described below.

b) Percentage of correct classes chosen - After a subjective evaluation of whether the classes chosen were correct or not, this criterion indicates what percentage of the total number of classes chosen for the sample or test set are correct. After a number of discussions, FTD decided that this was the primary evaluation measure in terms of their needs.

c) Document accuracy - A document is deemed to be classified correctly if at least one of the classes chosen by the sequential classifier is correct, based on a subjective evaluation of the document. The total number of correctly classified documents in a sample or test set is expressed as the percentage document accuracy.

The higher the value for each of the above evaluation criteria, the better the performance for the sequential classifier on the sample or test set being analyzed. Results will be reported in Section 4.3 in terms of these criteria.

4.2 Sample vs. Test Documents

As mentioned in Section 2.2, the sequential classification method requires a set of keywords representative of each of the defined categories or classes to be assigned to the documents, as well as a priori probabilities for all keywords within each category. The keywords and their probabilities are obtained from a representative sample set of documents as described in Sections 3.4 and 3.5.

The sample set thus represents a learning set from which the sequential classifier "learns" the selection criteria (i.e., the keywords and their a priori probabilities) to be used as a basis for choosing the classes which

are to be assigned to a document. After the selection criteria have been learned from the sample set, the sequential classifier is tested on an independent set of documents which were not used as part of the learning set. This test set is thus used in conjunction with the evaluation criteria described in Section 4.1 to evaluate the performance of the sequential classifier which to an appreciable extent, depends on how well the sample learning set represents the entire data base. If the sample set adequately represents documents from the data base, then the evaluation results should be comparable for representative sample sets and test sets which are processed by the sequential classifier under a given set of conditions as will be shown in Section 4.4.

4.3 System Parameters for the UDC Study

As reported in our earlier work on the sequential analysis model for automatic document classification [7,11], there are a number of user controllable parameters which may be used to fine-tune certain aspects of the performance of an actual implementation. These parameters, T, R, α, and δ, are introduced briefly here for the purpose of reporting the results and overall evaluation of the Phase I study in the next section. They will be discussed in more detail in Chapter 6.

The parameter T represents the initial set of keywords to be read from a document to preclude making a precipitious decision on the basis of the first few keywords. No decision regarding the classification of a document is made until at least T keywords are read. The parameter R controls the number of keywords read at each subsequent stage in the sequential classification process. R keywords must be read between each classification step. This parameter can be used to save computation depending on the quality of the keywords and the nature of the data base. This parameter is usually used in conjunction with T to influence the overall quality and efficiency of the sequential classification method.

The third paramter, α, represents the threshhold value which each class a posteriori probability must exceed for that category to be retained at each stage of the sequential classification process. Categories for which the a posteriori probabilities are lower than this prespecified threshhold are dropped until only appropriate classes are left. This decision will depend upon the termination criteria for the sequential algorithm to be described in Chapter 6.

The fourth parameter, δ, is the default probability - a small value which replaces a zero a priori probability used in the Bayes rule calculation. These default values are assigned to preclude early elimination of a class just because a keyword is read which happens to have a zero frequency count for that class.

Several different values for each of the above parameters were experimented with in the Phase I study until an improved set was obtained. The values used for the majority of the results reported in the next section

were as follows: T = 6, R = 1, $\delta = 5 \times 10^{-6}$ and $\alpha = 0.001$. Experiments using values other than these will be appropriately identified.

4.4 Evaluation of the Phase I UDC Study

In Section 3.3, it was explained that initially 1,837 sample documents were analyzed in order to get some tentative classification data and subsequently additional sample documents were analyzed to bring the total to 9,563 documents. Of all system parameters, the effect of this change was most pronounced in improving classification results.

In presenting the following data, there were two test sets of 100 documents each, which were randomly selected from documents not in any sample set; these are denoted Test Sets #2 and #5. From the 1,837 document sample sets, two sets of 100 documents were randomly chosen, denoted Sample Sets #1 and #4. When the additional 7,726 sample documents were analyzed, Sample Sets #11 and #14 of 100 documents each were randomly chosen from this group. Additional test sets were prepared, but the results presented here show that there is sufficient agreement between the several test sets and the several sample sets that evaluation of further test documents was not required. Also these results were interpreted and evaluated by FTD personnel, so the number of documents to evaluate had to be kept to a manageable size.

Table 4.1 shows classification results for the 1,837 document samples evaluated according to the criteria discussed in Section 4.1. The original objective was to try to match the author-assigned UDC codes as often as possible, but FTD indicated that these author-assigned UDC codes were too general or unreliable to use as the primary evaluation criterion. Instead a general agreement was made that % correct classes assigned would be a better indicator. In Table 4.1, note that the classification was much better for sample documents than for test documents. This indicates that the frequency distributions for keywords obtained using the 1,837 sample documents were not stabilized because this sample set was too small. Table 4.2 tabulates evaluations made for all 9,563 sample documents, and notice here that one cannot tell much difference between the sample or test documents in terms of classification performance. There are a sufficient number of sample documents for the keyword frequency distributions to stabilize. This observation will be important in the selection of final CIRC II classes described in Chapter 5.

In Table 4.1 two sets of keywords were evaluated for Test #2 and #5 documents. As indicated in Section 3.5 on keyword selection, when the number of keywords was cut substantially from 3319 to 1990, the % of the classes correct decreased, even though there was improvement in the other two criteria. That is, when the 1990 keywords became more specific, the author-assigned UDC code could be matched, and more often at least one class assigned by the algorithm was a primary subject of the document. But with fewer keywords, clearly more classes stayed in contention, slightly more being wrong than correct, and the % of classes correct decreased. Experimentation showed that T = 6 was required to obtain even these results for the

1990 keyword set. The lesson learned here was that the keyword set cannot be reduced too much without adverse effect on classification. For a description of how this reduction was effected, see Section 3.5.

DOCUMENT SET	T,KW SET	MATCHED ASSIGNED UDC CODE	NUMBER OF CORRECT CLASSES	NUMBER OF INCORRECT CLASSES	% CORRECT CLASSES	DOCUMENT ACCURACY
Sample #1	4,3319	87%	194	109	64%	97%
Sample #4	4,3319	90%	166	84	66%	93%
Test #2	4,3319	36%	135	133	50%	76%
	6,1990	52%	239	261	48%	82%
Test #5	4,3319	40%	126	110	53%	77%
	6,1990	52%	215	276	44%	88%

TABLE 4.1
Classification Results for the 110 UDC Classes
Using 1,837 Sample Documents ($\delta = 5 \times 10^{-4}$)

Table 4.2 shows substantially improved results using revised keyword frequency distributions from the larger 9,563 sample document set. A 3333 keyword set was obtained, and it is interesting to note that it differed only marginally from the 3319 keyword set used in Table 4.1. Further modifications of the 3333 keyword set did not appreciably improve the classification performance, which illustrates a remarkable stability in terms of keyword selection for this process. Other experiments to be described later will show that $T = 6$ and $\delta = 5 \times 10^{-6}$ were best choices for these parameters. For example, the decreased δ value is responsible for the decreased number of classes chosen, and the improvement in % classes correct. Note how similar the performance is for both sample and test documents, but for the sample documents, the number of matched UDC code is decreased as a sort of penalty for the more generally applicable classification method.

DOCUMENT SET	MATCHED ASSIGNED UDC CODE	NUMBER OF CORRECT CLASSES	NUMBER OF INCORRECT CLASSES	% CORRECT CLASSES	DOCUMENT ACCURACY
Test #2	63%	127	38	78%	86%
Test #5	64%	124	51	70%	88%
Sample #11	61%	121	39	76%	88%
Sample #14	57%	126	49	72%	87%

TABLE 4.2
Classification Results for the 110 UDC Classes Using 9,563 Sample Documents ($T = 6$, $\delta = 5 \times 10^{-6}$, 3333 KW)

Table 4.3 shows an evaluation by FTD of some of the same documents analyzed in Table 4.1. Recall that correct classes and document accuracy are determined subjectively in Table 4.1 by this investigator, whereas the results in Table 4.3 are determined subjectively by an information specialist of FTD. Note that the % of classes correct is much higher according to FTD, but the "primary class compatibility" is much lower than document accuracy, to which it is roughly equivalent.

Table 4.4 shows improved classification results for the sequential algorithm, especially for Test Set #5. Further improvement in the results for Test Set #2 will be indicated in the next section using a special technique. For the data of Table 4.4, it was noticed that the % classes correct criterion can be improved by selecting only the best classes, and thus selecting fewer total classes. A study is given in the next section to verify this observation. One approach is to arbitrarily choose the two classes with the highest confidence level α_i. The other approach is to select all classes whose confidence level α_i exceeds 0.1. Both approaches seem equally effective for Test Set #2, a trade-off occurring in the other two criteria. For Test #5, however, the more systematic selection method yields a higher % classes correct. In Chapter 6 it will be indicated how this is integrated into the termination rule of the sequential algorithm.

DOCUMENT SET	NUMBER OF CORRECT CLASSES	NUMBER OF INCORRECT CLASSES	% CORRECT CLASSES	PRIMARY CLASS COMPATIBILITY
Sample #1	119	41	74%	47%
Sample #4	151	25	86%	44%
Test #2	139	25	85%	83%

TABLE 4.3
Classification Results for the 110 UDC Classes
as Evaluated by FTD

DOCUMENT SET	SELECTION CRITERIA	MATCHED ASSIGNED UDC CODE	NUMBER OF CORRECT CLASSES	NUMBER OF INCORRECT CLASSES	% CORRECT CLASSES	DOCUMENT ACCURACY
Test #2	Top 2 α's	71%	130	62	68%	83%
	All $\alpha \geq 0.1$	63%	103	49	68%	87%
Test #5	Top 2 α's	71%	141	51	73%	95%
	All $\alpha \geq 0.1$	68%	114	27	81%	92%

TABLE 4.4
Classification Results for the 110 UDC Classes
Varying the Final Selection Criterion
($T = 6$, $\delta = 5 \times 10^{-6}$, 3333 KW)

4.5 Effects of System Parameters on Results

In the course of an investigation of keyword selection techniques, a number of selection algorithms and the resulting keyword sets were evaluated in terms of their effect on classification. One such keyword set is 3309 words, described in Section 3.5. Table 4.5 indicates a comparison between this set and a 3010 keyword set. Note the strong similarity between the results for the 3309 set and the 3333 set in Table 4.4. The 3010 keyword set is clearly inferior in every criterion category.

DOCUMENT SET	KEYWORD SET	MATCHED ASSIGNED UDC CODE	NUMBER OF CORRECT CLASSES	NUMBER OF INCORRECT CLASSES	% CORRECT CLASSES	DOCUMENT ACCURACY
Test #2	3010	58%	98	58	63%	77%
	3309	65%	111	50	69%	86%
Test #5	3010	65%	103	37	74%	86%
	3309	71%	114	33	78%	91%

TABLE 4.5
Classification Results for the 110 UDC Classes
for Two Keyword Sets ($T = 6$, $\delta = 5 \times 10^{-6}$)

A study of the variation of the default parameter δ is summarized in Table 4.6. The primary effect of an increase in δ is that more incorrect classes are allowed to be retained without any appreciable increase in the number of correct classes retained. If δ were decreased further, the problem is that a precipitous decision may be reached as the sequential process becomes quite unstable and too dependent upon initial keywords in the document. In addition, underflow problems will be encountered.

δ	MATCHED ASSIGNED UDC CODE	NUMBER OF CORRECT CLASSES	NUMBER OF INCORRECT CLASSES	% CORRECT CLASSES	DOCUMENT ACCURACY
$5x10^{-6}$	71%	114	33	78%	91%
$1x10^{-4}$	73%	113	44	72%	91%

TABLE 4.6
Classification Results for Test Set #5 for the
110 UDC Classes When δ is Varied (3309 KW, T = 6)

4.6 Effect of Dropping Classes from Consideration

As currently implemented, the sequential algorithm drops a class from consideration when its confidence level α_i falls below the specified threshold α after T keywords are read. For most of the experiments described here, T has been set at six keywords and α at 0.001. One might wonder whether better performance could be obtained if the classes were not dropped. Two experiments reported in this section deal with this question.

The first approach is to set T artificially high at T = 20. Then all classes remain in contention until 20 keywords are read or the end of the document is encountered. Table 4.7 shows the results of this study where a class is selected if its confidence level exceeds 0.9. The classes chosen are primary, if it describes a main subject of the document, secondary if it is a peripheral subject of the document, or incorrect if inappropriate for the document. Notice that for both Test Sets #2 and #5, there was virtually no difference in the performance for T = 6 and T = 20.

DOCUMENT SET	T	PRIMARY CLASSES	SECONDARY CLASSES	INCORRECT
Test #2	6	83	5	12
	20	84	4	12
Test #5	6	81	8	11
	20	81	7	12

TABLE 4.7
Classification Results for the 110 UDC Classes When
T Parameter is Varied ($\delta = 5x10^{-6}$, 3333 KW)

The second approach in the study of the effect of not dropping classes from consideration is summarized in Table 4.8. Here the entire document is read in one instance, a sequential decision with T = 6 and $\delta = 5x10^{-6}$ in the other, and a class is selected if its confidence level exceeds 0.1. Notice

that there is almost no difference at all in the performance for Test Set #5, but there is an appreciable variation for Test Set #2. In Chapter 7 it will be shown that Test Set #2 has some curious properties which may be involved here. The upshot of this study, however, is to conclude that dropping classes can be beneficial in that

a) some efficiencies can be effected in that fewer classes need be considered as more text is processed, and

b) at any stage, the classes remaining represent a partial solution to the question of what classes are applicable.

DOCUMENT SET	CLASS SELECTION CRITERION	MATCHED ASSIGNED UDC CODE	NUMBER OF CORRECT CLASSES	NUMBER OF INCORRECT CLASSES	% CORRECT CLASSES	DOCUMENT ACCURACY
Test #2	sequential	63%	103	52	66%	82%
	classes kept in	73%	114	36	76%	88%
Test #5	sequential	70%	111	31	78%	90%
	classes kept in	72%	110	34	76%	91%

TABLE 4.8
Classification Results for the 110 UDC Classes When Classes are not Dropped, Choose Classes $\alpha_j \geq 0.1$ (Sequential T = 6, $\delta = 5 \times 10^{-6}$), 3333 Keywords

4.7 Selecting Classes Early

A question arose as to whether improved performance could be obtained if a class were chosen early if its confidence level were high, since it might be lost by the time the termination condition is applied. In Table 4.9, it is shown how the addition of a class with $\alpha_j > .7$ after T = 6 keywords would affect the classification results.

It can be seen from the table that while this approach can be used to produce a few additional correct classes, it also produces at least an equal number of incorrect classes. This will definitely degrade the criterion % classes correct, and thus was not implemented into the final sequential algorithm.

DOCUMENT SET	KW SET	DOCUMENTS WITH NO EARLY CHOSEN CLASS	CLASS CHOSEN EARLY, BUT ALSO AT END	NEW CLASS CHOSEN PRIMARY-SECONDARY-INCORRECT
Test #2	3309	35	52	4-1-8
	3333	34	51	5-1-9
Test #5	3309	27	54	8-2-9
	3309 ($\delta = 10^{-4}$)	40	52	2-2-4
	3333	25	59	5-3-8

TABLE 4.9
Classification Results for the 110 UDC Classes When a Class is Chosen Whenever $\alpha_j > .7$ ($T = 6$, $\delta = 5 \times 10^{-6}$, and at end, select classes with $\alpha_j \geq .1$)

CHAPTER 5

DEVELOPMENT OF THE 98 CIRC II CLASSES

5.1 Expansion of the COSATI Classes

The COSATI classes were identified in Table 1.1 in Chapter 1. It was decided by FTD that the final CIRC II classes should be based upon an expansion of these classes, since a number of these classes identify major areas of interest of FTD users regardless of the number of documents in these areas.

This point is illustrated in Appendix A, where the distribution of COSATI codes is given over all documents disseminated during January through May 1976. There are three files for which statistics are reported, where the number of documents in each file are approximately three thousand, one hundred and thirty thousand, and five thousand, respectively.

The first file documents should substantially resemble the open literature documents discussed before, representing the preponderance of the documents disseminated. The design of the final CIRC II classes used these statistics in order to achieve near uniform distribution of documents over subject classes. The other files show much more emphasis in other categories: 1 Aeronautics, 15 Military Sciences, 16 Missile Technology, 17 Navigation, Communications, Detection, and Countermeasures, and 22 Space Technology, as might be expected. Thus these classes should be stressed out of proportion to their statistical representation in the entire data base.

These observations have motivated the design of the 98 CIRC II classes, presented in Appendix F. Notice that these classes preserve their identity with the COSATI classes as much as possible in their numbering from 1 to 98. Yet notice the overlap of these classes with the UDC classes in Appendix B, especially corresponding to COSATI class 9 (electrical and electronics), COSATI class 11 (materials), COSATI class 13 (mechanical engineering), and COSATI class 20 (physics), since these were the most successful UDC classes in terms of partitioning open literature documents.

There was still concern that there might not exist sufficient documents to justify certain areas as a CIRC II class. So retrievals were made by FTD on these subjects to resolve this issue. The following retrievals are typical of the way the decisions were made which areas should be retained as CIRC II classes, and which subject areas should be merged with other areas. Out of 690,000 documents, the following were retrieved documents based on the indicated search terms, illustrating a bona fide class:

Glass	9288	1.3%
Clay	1201	1.0%
Ceramics	3480	
Refractory	1968	
Cement	3303	1.5%
Concrete	7203	
Welding/Soldering	8835	1.3%
Motor/Motors	15000	2.2%
Crystal/Diffraction	6000	0.9%
Paper (with Timber)	7900	1.3%
Pulp (with Timber)	758	
Timber (with Wood)	247	

The following were shown not to constitute classes by themselves, and so were merged with other areas:

Isotopes	4021	0.6%
Solar Energy	476	0.07%
Textiles	3862	0.6%

This illustrates how initial decisions about class boundaries were either verified or challenged by retrievals into the actual CIRC II data base.

5.2 Manual Selection of Sample Documents

It was observed from early experiments that good classification results depend upon providing a representative sample set, in that keywords and keyword frequencies are determined from this data. Thus the sample documents must be carefully selected, and manually screened. In general, the selection procedure used the UDC parser described in Chapter 3, but occasionally incorrect documents were obtained here. The other methods of obtaining sample documents were by concept search terms and COSATI code. The documents obtained in this way had to be manually screened even more thoroughly.

This manual screening represented a major investment in effort. Each document examined was placed in one or several classes, or if it was not very representative or suitable as a sample document, it was deleted. For example, a document with poor keyword content or very short length would be deleted.

In Chapter 3 it was reported that about 10,000 sample documents were used to define the 110 UDC classes. In order to process about the same

number of documents, the goal was 15,000 sample documents, or about 150 sample documents for each of the 98 classes. In addition to excessive processing time, there is also a problem with frequency count overflow if more documents than this were analyzed.

5.3 Analysis of the Sample Documents by KEYFINDER for the CIRC II Classes

Sample documents for 87 of the 98 CIRC II classes were analyzed by KEYFINDER. Table 5.1 shows the number of documents selected in each case. This number was determined by the diversity of documents in the class and also the availability of good documents. Over 30,000 documents were manually examined and screened in order to select these characterizing documents for the 87 classes. Since this was the final processing of these sample documents, it was extremely important that the documents be selected carefully. Documents are not yet available for the following CIRC II classes, but will have to be analyzed subsequently to define these categories.

CIRC II Class	Description
16	R&D
72	MIS-TECH
73	MIS/SYS
79	NUC/MAT
80	NUC-REACT
81	NUC-PHYS
92	PROPEL
95	ECON
96	BUS
97	GOV/POL
98	SOC-SCI

A total of 13,358 documents were analyzed, and 1,066,992 tokens obtained which were not on the 1080 word stoplist. The total number of distinct words was 52,761, from which keywords were to be selected. Table 5.2 indicates the distribution of words over the CIRC II classes for which each occurred in at least one sample document from that class. This might be compared to Table 3.2 for the 110 UDC classes; clearly a better distribution has been obtained for the 87 CIRC II classes.

Returning to the data in Table 5.1, some final recommendations should be made to FTD in terms of where additional documents are required to be analyzed by KEYFINDER. Although the target figure for the number of sample documents for each class was 150, it does not necessarily follow that every

CIRC II CLASS		DOCUMENTS ANALYZED	CIRC II CLASS		DOCUMENTS ANALYZED	CIRC II CLASS		DOCUMENTS ANALYZED
1	AERO	137	34	EMAGTECH	166	67	LAB-TEST	109
2	AIRCRAFT	187	35	POWER	156	68	G-MIL	154
3	AG	228	36	MOTORS	149	69	MIL-MAT	138
4	LIVESTOCK	106	37	BATTERY	115	70	MIL-OP	166
5	ASTRO	147	38	FURNACES	155	71	CBR/NUC	119
6	ATMOS	191	39	OIL/LUB	108	72	MIS-TECH	---
7	BIO	135	40	CERAMICS	168	73	MIS/SYS	---
8	BACT	141	41	GLASS	107	74	NAV/GUID	146
9	PHARM	176	42	CEMENT	124	75	DETECT	129
10	ILL	101	43	PAINTS/CTG	169	76	CTRMEAS	82
11	MED/SCI	95	44	NF-MET	137	77	TELCOM	172
12	CLINIC	132	45	F-MET	147	78	RADIO	156
13	PHYS	164	46	WOOD	136	79	NUC/MAT	---
14	MED-INST	120	47	TEX/FIB	138	80	NUC-REACT	---
15	PSYCH	107	48	RUB/PLAS	154	81	NUC-PHYS	---
16	R&D	---	49	MATH	157	82	ORD	124
17	CYBER	144	50	CONSTR	180	83	MECH	132
18	CH-ENG	175	51	AIRC/HEAT	131	84	GAS/FL	139
19	PCHEM	183	52	ENGINES	122	85	VIB/ACOUS	124
20	ANALY-CH	173	53	TRANS	167	86	OPTICS	222
21	INORG-CH	172	54	CIV-ENG	159	87	THERMO	140
22	ORG-CH	196	55	PLANT-ENG	191	88	SOL-STATE	105
23	OCEAN	147	56	FOOD	170	89	EMAG	140
24	GEOG	129	57	FORGE	133	90	CRYSTAL	130
25	GEOPHY	188	58	MTL-HANDLE	173	91	FUELS	155
26	GEOL	147	59	ROLL/PIPES	176	92	PROPEL	---
27	MINE	212	60	MACH-TOOLS	187	93	SAT	148
28	PETROL	131	61	POWER-TRANS	167	94	SPACE	184
29	EL-INSTR	240	62	FLUIDS/PUMPS	206	95	ECON	---
30	EL-COMP	181	63	NAV-ENG	174	96	BUS	---
31	CPTR-HD	186	64	ENV-ENG	197	97	GOV/POL	---
32	CPTR-PG	151	65	WELDS	148	98	SOC-SCI	---
33	ELECTRONICS	192	66	MIL-TEST	163			
					TOTAL SAMPLE DOCUMENTS			13,358

TABLE 5.1 Sample Documents for the CIRC II Classes

class requires this many documents for adequate definition. For example, consider the following classes:

CIRC II Class	Documents Analyzed
39 OIL/LUB	108
88 SOL-STATE	105
90 CRYSTAL	130

CIRC II CLASS	WORDS INDICATING THAT CLASS	CIRC II CLASS	WORDS INDICATING THAT CLASS	CIRC II CLASS	WORDS INDICATING THAT CLASS
1	3603	34	3502	67	2972
2	6246	35	3184	68	5528
3	5075	36	2504	69	4135
4	2819	37	2736	70	5425
5	2584	38	2915	71	3634
6	4187	39	2457	72	0
7	4104	40	2475	73	0
8	4625	41	2153	74	4627
9	5356	42	2256	75	3472
10	3742	43	3402	76	2492
11	3636	44	2392	77	3756
12	4818	45	2609	78	3618
13	5051	46	2517	79	0
14	3207	47	2695	80	0
15	3224	48	3808	81	0
16	0	49	1906	82	3895
17	2964	50	2912	83	2359
18	3539	51	2924	84	2517
19	2872	52	3214	85	2147
20	3354	53	4074	86	3987
21	2763	54	2583	87	2435
22	3406	55	4747	88	2044
23	4555	56	5133	89	2249
24	2883	57	2393	90	2154
25	3598	58	2650	91	3958
26	3599	59	2777	92	0
27	4963	60	3493	93	4281
28	4029	61	2591	94	4576
29	3487	62	3076	95	0
30	3313	63	6451	96	0
31	3454	64	6768	97	0
32	3713	65	2411	98	0
33	3352	66	2707		
			TOTAL OVER ALL CLASSES		302,797

TABLE 5.2 Distribution of Words Over CIRC II Classes for Which Each Occurred in at Least One Sample Document From That Class

Even though fewer than 150 documents have been analyzed in each case, the subject content of these three classes is sufficiently narrow and homogeneous so this definition is entirely satisfactory. On the other hand, very broad classes which are multiple-faceted require a larger number of defining documents. A few examples are:

CIRC II Class	Documents Analyzed
2 AIRCRAFT	187
3 AG	228
27 MINE	212

Some argument might be made for adding even more documents to these classes. But then this might cause word frequency count overflows and also cause distortions in the sequential classification algorithm if a few classes are defined by an unreasonably large number of documents.

Finally, the following recommendations should be shared with FTD in obtaining a final definition of the 98 CIRC II classes:

1) only a few documents have been collected for the eleven classes: 16, 72, 73, 79, 80, 81, 92, 95, 96, 97, and 98; these should be defined from start, possibly using other than open literature documents;

2) the AERO Class #1 seemed a bit weak, as the available documents did not deal with many aspects of aerodynamics;

3) a few more LIVESTOCK Class #4 documents might be provided, but this class is probably adequately defined;

4) many more BIO Class #7 documents should be processed, as this is an extremely broad class, missing many facets, including specific species of flora and fauna;

5) several of the medical classes were not as well defined as might be desired due to gaps in the available documents; for example, classes ILL #10, MED/SCI #11, CLINIC #12, MED-INST #14, and PSYCH #15 should be augmented with complementary documents; class CYBER #17 documents contained no information on artificial intelligence or bionics, but emphasized general computer processing applications;

6) the class #67 LAB-TEST is a particular problem; perhaps the defining documents here should be examined and more defining documents provided;

7) a careful examination of the classes #68 G-MIL, #69 MIL-MAT, #70 MIL-OP, and #82 ORD should be made; it may be that more documents should be provided which yield keywords not found in the documents selected;

8) a special problem exists in Class #71 CBR/NUC; more nuclear documents may have to be used to achieve a well-rounded definition here;

9) another special problem was encountered with Classes #75 DETECT and #76 CTRMEAS; for example, no infrared or ultraviolet detection documents were available;

10) a number of class changes may be desirable after the classification system is in use for a while; for example, OPTICS includes optical techniques, optical radiation, lasers, and photography -- this may span too many topics for a viable class; also the breakdown between artificial satellites #93 SAT and other space topices #94 SPACE may not be appropriate, and it may be desirable to redefine this or any other CIRC II class. In Section 7.5, it will be indicated how such class redefinitions can be accomplished.

CHAPTER 6

CLASSIFY--THE SEQUENTIAL CLASSIFICATION ALGORITHM

6.1 The Sequential Classification Approach

In Section 2.2, an overview of the sequential classification algorithm was presented. In this chapter, a more detailed examination of this algorithm will be made, including computational details. A computer program documentation of the IBM 360 basic assembler language (BAL) version of CLASSIFY is provided in reference [3], which gives even more detail of this document classification system.

In the sequential approach, only as much of each document to be classified is read until it can be classified into one or more categories. A word in the document is isolated, and compared to a list of keywords. If it is not a keyword, another word is isolated and read. If it is a keyword, then access is made to a frequency table to obtain its a priori probability within each category. As this is done repeatedly with successive keywords, an a posteriori probability is calculated for each class.

This a posteriori probability corresponds to a confidence level, and for each remaining class it is compared to some predetermined threshold. If it is less than this threshold, then the class is dropped. When a termination condition is achieved, all classes with sufficiently high confidence levels are assigned to that document. If the end of a document is encountered before the termination condition is achieved, the document is deemed unclassifiable.

The important variables in the classification process are the frequency statistics on word types from each category. Clearly not all word types need to be retained for effective classification, and computationally it would be impractical to do so. Ideally, keywords selected to represent the categories should occur in only one category. However, usually only a few words in any data base occur in just one category, and these words will certainly not occur in every document. Therefore the challenge is to utilize words which overlap into several categories, and to discern their function in this case by frequency counts by class.

6.2 Input Information for CLASSIFY

The sequential classification method assumes the availability of a given number of subject categories, a selection of keywords representative of these categories, and the a priori probabilities of all keywords within each category.

More specifically, these inputs come from KEYFINDER, described in Chapter 7 and reference [2]. The first input is the set of all single-word keywords (or first word of a compound keyword). These will be in the form of a hash table for rapid accessing. When a keyword is found in this hash

table, its frequency count distribution will be required. This is stored in the frequency table input, and is accessed directly by each keyword. As will be indicated in the next section, probabilities will be formed from these counts by normalizing by total frequency class counts, another required input. The other input files relate to the compound keyword capability. Briefly, compound keywords consist of either two or three adjacent words in the text, and introduce considerable logic problems in the program complexity. More information on these required input files is given in the computer documentation for CLASSIFY given in reference [3].

The primary advantage of storing keywords within hash tables is that searches can be concluded successfully by examining as few as one cell of the table, or only slightly over this on the average. The processing time was reduced several magnitudes over a binary search using other sorting methods. A 60-80% loading factor gave satisfactory results, with a hashing algorithm consisting of adding the first eight characters, two characters at a time. When collisions did occur for the keywords, Day's algorithm [4] was used for collision resolution. At 70% loading factor, only an average of two address probes were required.

Next consider the preparation of the keyword frequency table. Let D_j represent the subset of sample documents associated with class C_j, and t_{ik} denote the number of occurrences of keyword W_i in document d_k. Then the frequency of keyword W_i given that a document is in class C_j is

$$f(W_i|C_j) = \sum_{d_k \in D_j} t_{ik}. \qquad (6\text{-}1)$$

These frequencies are calculated for each keyword and stored in the frequency table. When the keyword a priori probability estimates are used in the α-test, they are normalized by calculating the following:

$$P(W_i|C_j) = \frac{f(W_i|C_j)}{\sum_{k=1}^{N} f(W_k|C_j)} \qquad (6\text{-}2)$$

where N indicates the number of keywords.

During classification, the keyword frequencies do not change, and thus the denominator in equation (6-2) is calculated only once. The a priori class probabilities, denoted q_j, are calculated as

$$q_j = \frac{\sum_{i=1}^{N} f(W_i|C_j)}{\sum_{k \in L} \left[\sum_{i=1}^{N} f(W_i|C_k) \right]}, \text{ for each } j \in L, \qquad (6\text{-}3)$$

where L denotes the set of indices of classes remaining. As indicated

previously for equation (6-2), the bracketed portion of the denominator of (6-3) does not change as document classification progresses. Yet L may become smaller as the α-test requires classes to be dropped from consideration. Thus the denominator of (6-3) will have to be recalculated, and q_j for the remaining classes must be correspondingly updated.

6.3 The Sequential Classification Algorithm

Equation (6-1) indicates how the counts are obtained for each keyword W_i for each class C_j stored in the frequency table. We are now prepared to give a detailed explanation of how the sequential classification algorithm can be applied to a document to be classified, and especially how the α-test is performed.

A document is read into a buffer, and words are read sequentially from the document. A word is read from the buffer, and is hashed into the keyword table. If not a keyword, another word is read from the buffer. If a keyword is detected, then the keyword class frequencies $f(W_i|C_j)$ are read from the frequency table. Whether or not the α-test is performed depends upon two input parameters, T and R. T is the number of initial keywords that must be read before the first α-test is conducted. R denotes the number of keywords to be read between subsequent α-tests. Both parameters T and R prevent a precipitous decision from being made, especially in the presence of misleading or "noisy" keywords which happen to occur near the beginning of the document.

Bayes rule has been used successfully in many areas of statistics, e.g., pattern recognition [6,10]. It allows one to use a priori class probabilities such as from (6-3), and a priori keyword frequencies, such as from (6-2), in order to obtain updated estimates of the probabilities of each class, based upon the keywords observed. These updated estimates are called a posteriori class probabilities, and Bayes rule becomes:

$$\alpha_j = \frac{P(W_{i_1} W_{i_2} \ldots\ldots W_{i_n}|C_j)\cdot q_j}{\sum_{k\in L} P(W_{i_1} W_{i_2} \ldots\ldots W_{i_n}|C_k)\cdot q_k} \quad , \tag{6-4}$$

where $\{W_{i_1}, W_{i_2}, \ldots\ldots, W_{i_n}\}$ represents the sequence of keywords read from the document thus far, and q_j and L are defined in equation (6-3). The problem is that we have no way of evaluating the probability of a given sequence of keywords. Assuming keyword independence in terms of context, proximity, and order, this probability can be evaluated as:

$$P(W_{i_1} W_{i_2} \ldots\ldots W_{i_n}|C_j) = \prod_{k=1}^{n} P(W_{i_k}|C_j) \tag{6-5}$$

Notice that the entire product in equation (6-5) need not be recalculated with each α-test, but only those factors need be multiplied corresponding to keywords since the last α-test was made. The calculations in (6-4) and (6-5) may produce numbers which are very small (i.e., lead to computer underflow). This problem is circumvented by scaling each factor, since such scaling factors will cancel out in equation (6-5). The appropriate scaling factor depends upon the relative values of keyword frequencies and the maximum number of keywords expected to be read in any document, but for the system developed here a scaling factor of 1000 was found to be satisfactory to prevent underflow.

Another problem which must be addressed in equations (6-4) and (6-5) is that some keyword may not occur at all in any of the sample documents of one or more class, and thus the corresponding frequencies in equation (6-1) are zero. Indeed this is to be expected, since a keyword may be associated with one class, or several classes, but it would be undesirable that a keyword be associated with high frequency for all classes, as this sort of keyword would be of dubious use for classification. A zero frequency or probability used in a pure form of Bayes rule for equation (6-4) would then preclude the class from being selected if such a keyword were detected. This is unrealistic for this task, however, since a keyword may occur near the beginning of the document which is either an incorrect indicator of the content of the document, or else indicates a different aspect of that document, i.e., another class. Instead of using the zero probabilities which arise in equations (6-1) and (6-2), a default probability δ is defined for all such cases to be substantially smaller than the smallest nonzero probability found by equation (6-2), but large enough to allow a class to stay in contention in the α-test to be described below. Experiments in Chapter 4 showed $\delta = 10^{-6}$ to satisfy these requirements.

The α-test consists of testing whether for α_j, the a posteriori probability of class C_j which still remains in contention,

$$\alpha_j \geq \alpha, \text{ for } j = 1, 2, \ldots, L, \tag{6-6}$$

for the L remaining classes, where alpha (α) is an input parameter to the sequential classification algorithm. If any of the remaining L classes fails this test, this means that the sequence of keywords, $\{W_{i_1}, W_{i_2}, \ldots, W_{i_n}\}$, read from the document so far, do not sufficiently indicate that C_j is a correct class, but do suggest that other classes are appropriate for these keywords and this document. If so, class C_j is dropped from subsequent consideration. If one or more classes are dropped, then the class a priori probabilities q_j in equation (6-3) are recalculated.

The parameter α is related to the probability that a document is misclassified. The confidence level, or a posteriori probability, of class C_j given a sequence of keywords read from a document is defined to be α_j in equation (6-4). α_j is then bounded from below by α, for otherwise the sequential test would have removed C_j from consideration. The situation with the choice of α, T, R, and δ is quite complex, but generally

if α is decreased, the accuracy tends to improve, since more keywords are read before classification is attempted. However, α cannot be decreased arbitrarily, for then more classes remain in contention and the entire document must be read, and no classification decision has been made.

In Chapter 4, good classification performance was obtained with

$$\alpha = 0.001, \qquad \delta = 10^{-6}$$

$$T = 6, \quad \text{and} \quad R = 1.$$

This experience will be reinforced in Chapter 10.

6.4 Confidence Levels and Termination Criteria

Using the α-test, the number of classes remaining monotonically decreases. It is necessary to reach a decision to select one or more classes as quickly as possible, or if the entire document has to be read, should a subset of the remaining classes be selected to characterize the document, or should the document be declared unclassifiable? The most important measure to aid in making these decisions in the confidence level α_j for each remaining class C_j, as it represents the a posteriori probability of being the correct class for that document given the n keywords thus far observed.

The trade off issue here is that on the one hand, as many of the classes selected should be correct as possible, and yet as many correct classes should be assigned to each document as can be achieved. Both of these criteria should be achieved as rapidly as possible, i.e., as little of the document should have to be read as possible. A number of experiments were designed to investigate this trade off. Essentially it was discovered that in order to maximize the % correct classes, only one class should be chosen for each document, and only then when the confidence level exceeds 90%. If additional classes were selected, the % correct classes was found to be somewhat reduced.

Another issue to be resolved is the number of classes to be selected. Experiments with CIRC II documents and the sequential classification algorithm showed that consistently one, two, or three classes could be accurately assigned, depending upon the document. If an attempt was made to apply more classes, a significant decrease in accuracy was always noted. As a design limitation, it was decided to limit the number of potential classes to be assigned to five, although the sequential algorithm would assign that many classes to documents only in rare instances.

The best termination criteria were found to be the following:

1) continue reading more of the document if more than eight classes remain; if the end of the document is reached before the number of remaining classes drops below that level, the document is declared unclassifiable;

2) if six, seven, or eight classes remain, read additional keywords until the same classes are encountered three times after α-tests; if so, or if the document terminates, output up to five classes whose confidence levels exceed 0.1;

3) if four or five classes remain, read additional keywords until the same classes repeat; if this occurs or if the end of the document is encountered, output any class whose confidence level exceeds 0.1;

4) if one, two, or three classes remain, retain any class whose confidence level exceeds 0.1.

Substantial experimentation showed that any class with confidence level less than 0.1 was a poor risk in that the overall % classes correct criterion substantially deteriorated. The choice of eight classes in (1) was not entirely arbitrary, for if more than eight classes remain, the chances of one or two classes having a substantial confidence level is considerably reduced.

A most important design consideration was to output confidence levels along with selected classes in the format indicated in Section 2.4. The user can then decide whether or not a chance should be taken on a class with confidence level at 0.1, 0.3, or 0.6.

Table 6.1 reports some termination criteria experiments. The termination criteria described by rules 1) - 4) above are utilized as a standard, and other stopping criteria compared to this. If all classes are selected whose confidence levels exceed 0.1 when eight or fewer classes are obtained, only a few more correct classes were selected, and the % classes correct dropped significantly.

The last two sets of data in Table 6.1 have criteria

a) only stop when a single class is left, or if end of document is reached, take the class of highest confidence level; and

b) stop whenever a class achieves a confidence level of 0.9, respectively.

These two criteria had substantially higher % correct classes criteria, but note that significantly fewer classes were chosen. When you add the consideration that much more processing was involved (many documents were read entirely), it is clear why criteria 1) - 4) was finally selected.

For nearly all documents, it was quite typical that confidence levels exceeding 0.85 were achieved before ten keywords were read.

6.5 Modifications to CLASSIFY

CLASSIFY can easily be modified in its operation either by input data or parameters. For example, keyword selection may exercise a significant effect on classification, and the user is free to choose this, as it is an

EXPERIMENT, DOCUMENT SET	MATCHED ASSIGNED UDC CODE	NUMBER OF CORRECT CLASSES	NUMBER OF INCORRECT CLASSES	% CORRECT CLASSES	DOCUMENT ACCURACY
Criteria 1)-4)					
Test #2	73%	114	36	76%	88%
Test #5	72%	110	34	76%	91%
Stop at 8 classes; take all $\alpha_j \geq 0.1$					
Test #2	72%	120	48	71%	90%
Test #5	73%	116	44	72%	91%
Stop at 1 class or end-doc; Select top α_j					
Test #2	69%	87	21	81%	84%
Test #5	68%	84	17	83%	83%
Stop at first class with $\alpha_j \geq 0.9$					
Test #2	66%	83	17	83%	83%
Test #5	62%	81	19	81%	81%

TABLE 6.1 Termination Criteria Experiments with 110 UDC Classes,

$T = 6$, $\delta = 10^{-6}$, 3333 KW.

input to CLASSIFY. Similarly keyword frequencies are important, and are input data to CLASSIFY. In this chapter, parameters T, R, α, and δ were defined, and their effect on classification discussed.

Another variable the user may have to change is the number of CIRC II classes, which currently is at 98. The CLASSIFY program is established so as to be able to handle up to a maximum of 110 classes. If someone chooses to add more classes, CLASSIFY documentation in reference [3] explains what minor program changes are required. If the number of CIRC II classes were changed to exceed 110, nearly all of the program arrays must be increased to accommodate this revision, and the user must be careful about such a change.

CHAPTER 7

KEYFINDER - THE SAMPLE DOCUMENTS ANALYZER

7.1 The KEYFINDER Software

The KEYFINDER software system is designed to set up all the input tables required by the CLASSIFY program. This is done by obtaining frequency counts of all non-stoplist words from a set of sample documents. The frequency distributions of these words over all classes are then examined, and the most promising words are selected as keywords. These keywords and their related class frequencies are the primary inputs which the CLASSIFY program uses to classify documents. These keywords are supplemented by a set of compound keywords each of which consists of two or three adjacent words, treated as a single keyword concept. The compound keywords are important when the words comprising the compound keywords would, taken by themselves, be ambiguous or contain information of little use in classification. Compound keywords are discussed further in Chapter 8. A computer program documentation of KEYFINDER is provided in reference [2], where a complete and detailed description of the software and how it operates is presented. The constituent parts of KEYFINDER will be considered in this chapter and described at a level to allow one to understand the purpose of that aspect of the software, and how it was designed to accomplish that objective.

The first major subprogram is KEYFIND, which is run once for each class to be defined. It accepts the hashed stoplist as input along with the set of documents (in CIRC II output format) which defines the class, the class number, the compound keywords, and some other parameter information related to the compound keywords. The KEYFIND program examines the input documents and outputs each word not on the stoplist along with the class number. If a compound keyword is encountered, its corresponding frequency record is updated.

In the second step of KEYFINDER, the words from the various runs of KEYFIND are combined and sorted using the IBM SORT/MERGE package. This can be done in stages if desired by using the sorted output from a previous run as one of the inputs to the current run.

The output from the SORT is combined by the next program (PHASE3) so that there is one record for each unique word. This record contains the word and the frequency counts for this word by class.

Next the program CONVERT takes the PHASE3 output along with the compound keyword frequency data, and creates the files required by CLASSIFY, including the selected keywords. Using the frequency data in each record, CONVERT determines whether or not the word should be selected as a keyword. If so, the frequency data for each selected keyword and for all compound keywords are prepared for the CLASSIFY program. Another set of data required is the total frequency by class summed over all keywords. As a final step, the sequential keyword file is hashed so that a hashed keyword table can be read directly by the CLASSIFY program.

7.2 Required Input for KEYFINDER

The sample documents to be analyzed by KEFYINDER were chosen to be in CIRC II output format rather than the IPIR format. The reason this was done is that sample documents had to be selected from documents existing in the data base rather than incoming documents in the IPIR form to be processed. These documents can be easily retrieved, and any selected fields printed and output to tape for further processing. All the documents analyzed by the final run of KEYFINDER were delivered as data with the software so that anyone could discover for themselves what documents were used to characterize each class. The method by which these sample documents were selected is described in Chapter 5. In order to keep the input to KEYFIND simple, only documents for one class are processed with each run, and this class must presently be identified as a class number between 1 and 98.

Notice that there is a different philosophy in the way simple keywords and compound keywords are handled by KEYFIND. For the simple keywords, a new file of token counts is always made, and sort-merged with previous counts. For compound keywords, however, the compound keywords must be a priori specified, and a table of class counts for each compound keyword is input and then updated with each run. With each run, new compound keywords can always be specified, but it should be emphasized that class counts cannot be made over documents already analyzed. However, it is always possible to rerun all sample documents through KEYFINDER, but this involves a large amount of processing, analyzing over 15,000 documents.

The final stoplist presently consists of 1080 words, each truncated to a ten character string. This can be easily changed by hashing the revised stoplist into a hash table, and this used as input to KEYFIND, but again this will not affect sample documents already processed.

7.3 SORT/MERGE and PHASE3 - The Sorting and Counting Functions

SORT/MERGE performs the sorting and combining functions for KEYFINDER on the individual occurrences of each non-stoplist word detected by KEYFIND. It should be emphasized that there will be many word token files - one produced by each run of KEYFIND. These files must be combined and sorted. The primary sort key is the word token itself with the records being subordered by class number and then document number.

It should be noted that the final sorted file from SORT/MERGE was delivered as data with the software. This was done so that as additional sample documents are analyzed by KEYFINDER, the resulting word tokens can be merged with this file to update keyword selections and frequency distributions. As discussed in Chapter 5, since eleven classes are not yet defined, and additional documents may be added for any of the other classes, this option will have to be utilized to produce a final operating system for all 98 CIRC II classes.

PHASE3 then performs the counting function on the sorted word token file from SORT/MERGE. PHASE3 produces a record for each distinct word in that file which consists of the word, its frequency count by class, together with some summary information concerning the counts.

A design decision was made to allow only one byte for each class count within each word, for otherwise the frequency table would require an excessive amount of storage. This decision was reasonable, for although 50,000 words over 87 classes were produced by PHASE3, only in 17 instances (for 15 distinct words) did the count overflow this byte, i.e., exceeded a count of 255. Table 7.1 shows these 15 words, together with the CIRC II class byte which overflowed, and the total count of that word for that class. In some cases, e.g., NAVIGATION or OIL, the total count is only marginally larger than 255. In this case, we could just terminate the count at 255, and the overall frequency distribution would not be affected significantly. However, in most cases, e.g., for AIRCRAFT and FUEL, this approach would give a distorted view of how important to classes 2 and 56, respectively, these words would be. Also, nearly all the words should be keywords for the overflow class, except possibly LIGHT in class 86.

WORD	CIRC II CLASS	TOTAL COUNT
AIRCRAFT	2	617
ENGINE	52	298
FOOD	56	686
FUEL	91	349
GLASS	41	317
LIGHT	86	315
MEDICAL	11	284
MEDICAL	12	323
MILITARY	68	300
NAVIGATION	74	260
OIL	28	288
PACKAGING	55	397
PACKAGING	56	309
SATELLITE	93	420
SPACE	94	438
VALVE	62	352
WELDING	65	347

TABLE 7.1 Those Words Where Class Counts Overflowed

A method must be found to remedy these overflows when they occur, for the overflows cannot be predicted ahead of time. In order to do this, we must return to the mathematical description of CLASSIFY given in Chapter 6. The principal computation affected is that of the a priori probabilities of each keyword by class, given as equation (6-2), repeated here as:

$$P(W_i|C_j) = \frac{f(W_i|C_j)}{\sum_{k=1}^{N} f(W_k|C_j)} \qquad (7\text{-}1)$$

where W_i is assumed to be the word which has overflowed in the frequency count $f(W_i|C_j)$ for class C_j, and we will consider the selection of N keywords using these frequency counts.

The problem is schematically indicated in Figure 7.1, where the frequency count data can be visualized as a table, with rows corresponding to distinct keywords obtained from PHASE3, columns corresponding to the CIRC II classes, and entries of the table corresponding to the frequency count of each word by class. Assume the count of word W_i in class C_j has overflowed, so the j^{th} column sum corresponds to the denominator of equation (7-1), and the numerator to the overflowed count.

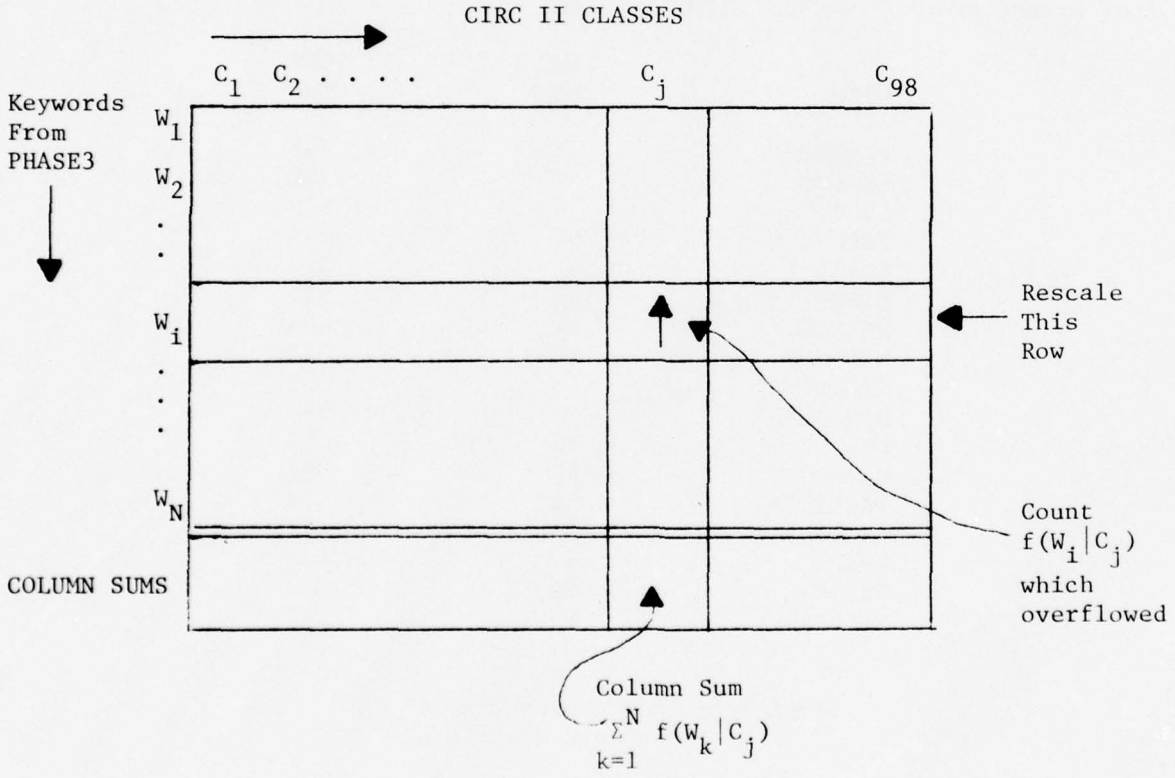


FIGURE 7.1 Frequency Table in Which the Count for Word W_i has Overflowed for Class C_j

The solution to the overflow problem we propose is to scale the i^{th} row by multiplying each entry by the factor

$$\frac{255}{f(W_i|C_j)}$$

and then rounding up, making sure the ij^{th} table entry is 255. Notice that for the i^{th} row of the frequency table, given that an overflow has occurred, it represents the best we can do to readjust the counts, and yet try to retain the distribution of counts across classes for that word W_i. Some errors have been introduced, however, that must be considered:

1) the modified counts might affect keyword selection; but since all the keyword selection criteria described in Chapter 3 and the next section examine only counts within each word, there should be negligible effect on keyword selection;

2) there is rounding error in the i^{th} row frequency entries; however, since the maximum count is known to be 255, then relative to this in a probability calculation, it matters little whether a count of one, two, or three is obtained; note, however, a count of zero might make a difference, and this is why we propose rounding up;

3) all the column sums

$$\sum_{k=1}^{N} f(W_k|C_m)$$

change for each class C_m, including class C_j; here we argue that these sums are so large that rescaling the i^{th} row entries should exert but a minor perturbation; if any of these column sums were changed to any degree, then we might have to reexamine this approach, but it would also say that there are clearly insufficient words which primarily represent that class;

4) notice that the class a priori probabilities q_k all change as defined in equation (6-3) for all classes C_k; this change is primarily due to variation in column sums, which is dealt with in 3).

In summary, then, the final frequency table from PHASE3 has been rescaled for the 15 distinct words in Table 7.1. Note that if documents are added to define the additional classes, almost certainly some keyword frequency count will overflow for that class. In this case, rescaling will again have to be applied. It should be clear that in this situation it would be best to have an unscaled frequency table, and rescale in order to correctly take into account any modified counts. Thus both unscaled and scaled frequency table data will be delivered with the software.

7.4 CONVERT - Selection of Keywords

The CONVERT program algorithmically selects which words from PHASE3 are to be keywords based upon the frequency counts of the words. It then sets up the files required by the CLASSIFY program.

The input data required by CONVERT is:

1) the compound keywords and their associated frequency counts over classes;
2) the frequency table for single words;
3) sequential files of "keep" words and "throw" words.

The reason for the keep and throw lists is the following. The keyword studies in Chapter 3 illustrated that automatic keyword selection techniques can never guarantee that an intuitively poor keyword will not be chosen, or that an intuitively good characterizing keyword for a class will not fall just below a selection threshold. Thus after sufficient keyword studies, a keyword which should be included even though it tends to be eliminated by automatic keyword criteria is put on the keep list. Similarly, a word which persists in passing the automatic keyword criteria, even though it clearly should be deleted as a keyword, is put on the throw list.

For each word, the following processing is done using the frequency table. If it is the first word of a compound keyword, a flag is set to indicate this. After this, a check is made to see if the word is on either the keep list or throw list. If it is on neither of these lists, the frequency data is tested to see if the word passes the tests for inclusion in the keyword set. If the word is on the throw list or if it fails one or more keyword tests, it is rejected as a keyword. If the word is on the keep list or if the word has passed all of the keyword tests, it is retained as a keyword along with its frequency data.

It should be noted that the automatic keyword criteria are modular, and can easily be modified if different keyword sets are required than the current selection criteria obtains. An output print routine is available to output the selected keywords by class, including the most important classes which each account for more than 10% of the total frequency count of that keyword. Thus a keyword may be printed a number of times, being repeated for several classes. The keywords associated with several typical classes are indicated in Appendix D, to illustrate the format of this printout.

7.5 Modifications of KEYFINDER

The KEYFINDER software is a tool to analyze sample documents to define classes, especially in an environment where the documents or classes may be made available piecemeal. Thus the software had to be developed to be as flexible as possible and can be modified to accommodate a number of changes. Specifically the following modifications have been allowed for:

1) the stoplist can be changed;
2) more classes can be added;
3) new classes can be defined by submitting input defining documents to KEYFIND;
4) additional documents can be submitted to further define an already existing class;
5) a class can be deleted;
6) new compound keywords can be added;
7) the keywords can be changed by modifying the keyword selection criteria, the keep list, or throw list.

The stoplist can be changed very easily, simply modifying the original set of input cards by inserting or deleting stoplist words. These must be hashed into a new hash table, which might have to be enlarged in order to keep the present loading factor for rapid search. The most serious problem is that counts of words cannot be changed for documents already processed by KEYFINDER. If this is required, it will be necessary to reprocess all sample documents by KEYFINDER.

The KEYFINDER software presently assumes at the input stage there are 98 classes. If classes are to be defined within this range (for example, eleven such classes are yet to be defined), no modifications have to be made at all; the characterizing documents need only be submitted to KEYFIND. If a new class beyond 98 is to be defined, only one input parameter need be changed, and the system extends in a simple way. It was decided with FTD that a reasonable expansion capability would be up to 110 classes. If more than 110 classes are to be defined, substantial modifications to KEYFINDER are required including an expansion of the frequency tables.

If a class has already been partially defined by documents submitted to KEYFINDER, but it is desired to define alternate aspects of that class by other documents, they can be simply submitted to KEYFINDER with the class identified. These documents will be analyzed, word tokens merged with the others from this class, and the counts updated.

There are more logical difficulties if an existing class is to be deleted, or must be modified so that previously analyzed documents for this class are to be deleted. The simplest way to accomplish this is to do a search of all word token output of SORT/MERGE, and delete all occurrences of word tokens from that class. That class can then be redefined with new documents, and the new word tokens will be merged with the modified token file. Further processing will produce keywords for an entirely new class.

New compound keywords can be added whenever a run of KEYFINDER is made. The problem is that these new compound keywords were not searched for in previously analyzed documents, and thus final frequency count distributions are somewhat suspect. The only way around this problem is to completely analyze all documents over again after a final determination of all compound keywords has been made.

The CLASSIFY algorithm is very dependent upon the keywords selected, so given that the best sample documents have been analyzed by KEYFINDER, and the word frequencies determined, the only input affecting classification is the keyword selection. Thus CONVERT is very flexible, allowing a wide range of keyword selection techniques. A new keyword selection criteria can easily be inserted in place of the present one in CONVERT. Words can easily be added to or deleted from the keep list or delete list. The one operation which cannot be done at the CONVERT stage is to add more compound keywords, because they had to have been defined before some subset of documents were analyzed.

A study of these seven types of changes in the output of KEYFINDER shows that it is extremely flexible, and considerable thought has been given to its design to yield this flexibility. For further details, see reference [2].

CHAPTER 8

COMPOUND KEYWORDS

8.1 Definition of Compound Keywords

A compound keyword consists of two or three adjacent words, treated as a single keyword concept. Each constitutent word is assumed to consist of no more than 14 characters, or else it is truncated to that length. A design decision was made early that at most three adjacent words would capture nearly all compound concepts which would occur in practice. The problem is that this decision must also take into account the complexity and computation time of a more general approach, and recall that the CLASSIFY software must be fast in terms of the number of documents classified per unit time.

It should be emphasized that except for the three word limitation and adjacency restriction, any configuration with these constraints can occur. For example, any constituent word of a compound keyword can also be a keyword, including the first word. Furthermore, the first two words of any three-word compound keyword can also be a distinct compound keyword, and other compound keywords can be formed by adding any number of words after either a common first word or a common two-word pair. This sort of flexibility considerably complicated the design of the compound keyword software. For example, the following phrases could all be (compound) keywords in the system:

MILITARY
MILITARY HARDWARE
MILITARY HARDWARE DESIGN
MILITARY HARDWARE MAINTENANCE
HARDWARE DESIGN
COMPUTER
COMPUTER HARDWARE
COMPUTER HARDWARE MAINTENANCE

It should be noted, however, that all these concepts are not (compound) keywords in this system, as they have become far too specific for this data base classification.

A list of some typical compound keywords finally selected are given in Appendix E. 940 compound keywords were analyzed by KEYFINDER, but the number selected was reduced to 464 after reviewing frequency data, as some of these occurred too infrequently. It must be emphasized that many of these compound keywords are associated with classes not yet analyzed, and to be on the safe side, were retained for these classes. This was done so that when frequency counts are eventually obtained, a decision could be made as to which compound keywords to retain.

Without frequency data, it is difficult to determine how effective the compound keywords might be for classification. For example, there is no doubt that SOLAR ENERGY is a useful compound keyword, and does occur often

enough to justify its retention. But there are many compound keywords which appear to be useful, but never or hardly ever occur in sample documents. Thus no frequency distribution data can be obtained on these compound keywords. This is heavily dependent upon the way in which the sample documents were chosen. If sample documents were selected so as to fully develop a particular compound keyword concept, then they would not be representative within a set of only 150 documents for that class. Yet it was desired that the software have the compound keyword capability in case it was needed.

As the next section will show, compound keyword processing will lead to considerable complication in both the KEYFINDER and CLASSIFY programs. The run time of KEYFINDER is considerably longer with compound keywords, but this is not too serious as it is a one-time off-line program. If there are few compound keywords, then it should be emphasized that the run time of CLASSIFY will not increase too much, because the only additional work is to check a flag when a keyword is found in the hash table, and if this is not the first word of a compound keyword, nothing extra need be done. If, however, many compound keywords were to be detected, then the next section will indicate the extra complexities involved, which would definitely slow down the CLASSIFY program.

8.2 Construction and Use of the Compound Keyword Tables

In order to detect compound keywords, three tables must be constructed by KEYFINDER and utilized by the CLASSIFY software. The three tables are shown in Figure 8.1. The first table contains each unique first word of the set of compound keywords and for each, a pointer (TWOPTR) to the initial (or only) second word of this compound keyword. The second table contains all second words of compound keywords, and three pointers. The first pointer (THRPTR) indicates the initial (or only) third word, in the third table, for this compound keyword. The pointer is null if no third word exists for this pair. The second pointer (TABLPTR2) points to the frequency table if the first word-second word pair constitutes a compound keyword, and it is null

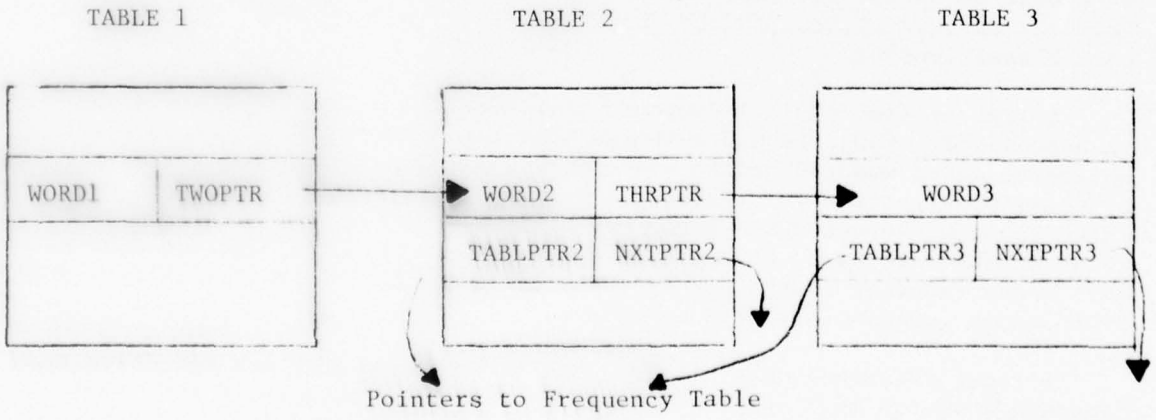


FIGURE 8.1 Three Compound Keyword Tables

if this pair is not a compound keyword. The third pointer (NXTPTR2) points within the second table to the next second word associated with this first word; if none, this pointer is null. The third table contains all third words of compound keywords and two pointers. The first pointer (TABLPTR3) points to the frequency table for this compound keyword triple. The second pointer (NXTPTR3) indicates within the third table the next third word associated with this first word-second word pair; if none, this pointer is null.

The tables are constructed by the following procedure. The compound keywords are lexicographically ordered, i.e., in alphabetical order by first word, then subordered by second word and then third word. Each unique first word is placed in TABLE 1 as it is encountered, and the pointer to the second word in TABLE 2 is entered at this point. The second and third words and all pointers except frequency table pointers are also entered as the compound keywords are read. The frequency table pointers and data are entered after all compound keywords are read.

Both KEYFIND and CLASSIFY search the tables as follows. When a word is found in TABLE 1, the next buffer word is checked against the TABLE 2 entries for the second word. Recall that all relevant entries in TABLE 2 are checked using the pointer NXTPTR2. If it is not there, the first word may be a single word keyword. If the second word is in TABLE 2, and there is no third word (if THRPTR is null), then the frequency count is accessed for this two-word compound keyword. If there is a third word, the next buffer word is checked, and if it matches a third word for this pair, this defines a three-word compound keyword. Recall that several entries may have to be checked in TABLE 3 using NXTPTR3. If there is no match, back up to TABLE 2, for the matched two words may still be a two-word compound keyword. If so, access the frequency table for it. If this fails, we back up to TABLE 1 and process it as a single word.

An example of the construction of these three tables might clarify this complicated situation. Suppose we read the following seven compound keywords:

1) A1 L2
2) B1 G2
3) B1 G2 H3
4) B1 G2 J3
5) B1 K2 Y3
6) B1 M2
7) C1 D2 X3

Figure 8.2 shows the construction of the three tables. A few words might be said about the size of these tables. TABLE 3 has size equal to the number of three-word compound keywords. The size of TABLE 1 is the number of unique first words, and the size of TABLE 2 must equal the number of unique first word-second word pairs.

	WORD1	TWOPTR
1	A1	1
2	B1	2
3	C1	5

TABLE 1

	WORD 2	THRPTR	TBLPTR2	NXTPTR2
1	L2	0	1	0
2	G2	⊠ 1	2	⊠ 3
3	K2	3	0	⊠ 4
4	M2	0	6	0
5	D2	4	0	0

TABLE 2

	WORD 3	TBLPTR3	NXTPTR3
1	H3	3	⊠ 2
2	J3	4	0
3	Y3	5	0
4	X3	7	0

TABLE 3

FIGURE 8.2 Compound Keyword Tables - an Example

The first compound keyword A1 L2 is entered into TABLE 1 and TABLE 2 respectively, with THRPTR and NXTPTR2 set to zero for the latter. The same occurs for B1 G2. Note when B1 G2 H3 is read, this calls for THRPTR for G2 in TABLE 2 to change, and an entry H3 is made in TABLE 3. From this point it should be fairly clear how the other entries are made or modified in the tables. The pointers TBLPTR2 and TBLPTR3 which point to the frequency table data are listed for clarity in the order in which the entries were made. Actually another pass is made through the tables to reset these pointers.

This example clearly shows the price in complexity which has been paid for the desired flexibility using compound keywords.

CHAPTER 9

CLASSIFICATION OF DOCUMENTS OF VARYING SUBJECT

9.1 Documents Which Change Subject

One type of document which may cause problems for the sequential classification algorithm is one in which the subject changes in the course of the document. Intelligence Report (IR) documents are potentially documents of this type, where a subreport on different technical subjects may be included within the same report. If the sequential classification algorithm is applied to this type of document, usually only the first subject area treated within the report will be detected, and this assumes the first subject will be discussed long enough to define sufficient keywords to be detected. If the subject area changes too rapidly, then no classification decision can be made, and the document would be declared unclassifiable.

Intelligence reports are identified by document accession codes, so that it was proposed that a different classification mode of classification could be used on this type of document. It is unfortunate that subject area changes do not always occur at new paragraph boundaries, so that there do not exist syntactic clues which can reliably predict where these changes occur.

In the next section, a technique is described which is proposed for classifying documents which change subjects.

9.2 Bayesian Distance Classifier

Consider m classes C_1, C_2, ...C_m into which an incoming document may be classified. At a given stage of the sequential process suppose we have read n keywords W_1, W_2, ..., W_n, and that we represent the effect of all these keywords by a single variable y. Then the a posteriori probability of a given class after observation y has been made can be represented as follows:

$$P(C/y) = [(C_1/y), \ldots, P(C_m/y)].$$

This will be called the Bayesian probability vector.

The Bayesian distance on the probability space of a set of classes after an observation y, denoted by $D_B(y)$, is represented by a magnitude and a direction, i.e., $D_B(y) \equiv [Mag,Dir]$, where Mag is equal to the squared Euclidian norm of the Bayesian probability vector and Dir is the index of the class having the highest a posteriori probability after observation y,

$$\text{Mag} = \sum_{k=1}^{m} [P(C_k/y)]^2, \qquad (9\text{-}1)$$

and Dir = i such that

$$P(C_i/y) = \max_k [P(C_k/y)], \; k = 1 \text{ to } m. \qquad (9\text{-}2)$$

It has been found that this Bayesian distance measure is a very sensitive indicator of a keyword which is not indicative of the primary class of the document. As several good keywords have been read, the Bayesian distance magnitude is found to increase monotonically, with the direction remaining constant, and indicating the correct primary class. Then when a spurious or "noisy" keyword is read, it is found that the Bayesian distance is very sensitive to this keyword, and either there is a precipitous drop in the magnitude of this measure when this keyword is included, or else the direction will switch to an entirely new class.

In the subsequent discussion, the class most indicated by a keyword, i.e., its largest a priori probability, will be denoted as its index for brevity. For example, for a given keyword W_i, with m = 5 classes, suppose the a priori probabilities for W_i are given as

$$[P(W_i|C_j)] = [0.1, 0, 0.5, 0.3, 0.1].$$

Then the index of keyword W_i is 3.

Other properties of this distance measure are reported in references [7.11], but the most important property is that the Bayesian distance is a very sensitive measure of the subject content of the document.

A Bayesian distance classifier has been developed which utilizes this Bayesian distance measure for document classification. It proceeds differently from the sequential algorithm, in that it chooses the most appropriate classes based upon a Bayes distance analysis of the keywords read, rather than eliminating inappropriate classes and finally selecting the correct classes, as in case of the sequential algorithm. The Bayesian distance classifier is less susceptible to spurious keywords in the beginning of a document than is the sequential algorithm, but the Bayesian distance classifier was rejected for overall classification of the CIRC II documents because the sequential algorithm is much more efficient, and the latter has met very severe efficiency and processing requirements.

The Bayesian distance classifier proceeds as follows:

1) three keywords are read from the document to be classified; the strongest class is tentatively identified, and all the keywords of that index are saved and their Bayesian distance calculated; the other keywords are set aside for possible later use;

2) subsequently one keyword is read at a time; if the index is the same as the direction of the current Bayesian distance, it is retained, otherwise it is set aside;

3) this is continued until either classification conditions are satisfied, fifteen keywords are read, or else the end of the document is reached; in order to classify the document as the direction of the current Bayesian distance, the classification conditions are:

i) at least 3 keywords have been identified with that index, and these have a Bayesian distance magnitude of at least 0.75; and
ii) either at least 6 keywords are in the selected set or the magnitude exceeds 0.9;

4) if either fifteen keywords are read, or the end of the document has been reached, and also no primary class has yet been assigned to the document, then the primary criteria are relaxed, and the most stringent magnitude criterion reduced to 0.7; either a primary class is thus selected, or the document is deemed unclassifiable;

5) there is a provision for examining the rejected keywords if a primary class cannot be found; this consists of essentially switching the two sets of keywords;

6) after a primary class is chosen, this class a priori probability is set to zero, which may allow other classes to be selected, as the effect of the primary class is thus essentially eliminated;

7) additional primary classes are obtained by reading keywords initially set aside, and then additional keywords read from the document;

8) after as many primary classes are chosen as possible, and all fifteen keywords have been read, selection of secondary classes is made; all the keywords are searched for additional classes which satisfy more liberal selection criteria.

9.3 Compound Documents

Although it was proposed that the Bayesian distance classifier be used for Intelligence Reports, a sufficient sample of unclassified Intelligence Reports was not available. Thus two sets of compound documents were produced from Test Sets #2 and #5 defined in Chapter 4 by concatenating three documents together as a single compound document. Each set consisted of 33 compound documents. These documents are thus known to change subject twice, and were used to evaluate the proposed approach for analyzing documents which change subject content. It should be emphasized that these were UDC related documents, and the classes to be assigned were the 110 UDC classes utilized in the studies reported in Chapter 4.

9.4 Experiments with Compound Documents

In order to apply the Bayesian distance concept to the problem of classifying compound documents, some changes were made in the method. The central search was for a primary class; when it was found, the classification process was reinitialized. Only when a primary class could not be detected within the fifteen keywords read (or the end of the document is encountered) is the secondary classification approach utilized. Thus the document is divided into natural blocks of text, consisting of blocks containing enough keywords to definitely yield a primary class. The same primary class may be obtained any number of times, but a new primary class should be chosen when the subject changes. Note that for this approach to make sense, the compound document, or any document to be analyzed for a change of subject, should be longer than most CIRC II documents. This appears to be a reasonable assumption.

Table 9.1 shows the results of experiments with this approach on compound document sets #2 and #5. For every document, at least one assigned class was correct, so this data is not reported in any of the results. Note that the % correct classes criterion could bear considerable improvement. The improvement can clearly be made by decreasing the number of incorrect classes by applying a more stringent criterion for a primary or secondary class to be accepted. Another reason for the poor performance was that documents associated with the general UDC classes were in dominance, and many of the classes judged to be incorrect were these general classes.

COMPOUND DOCUMENT SET	NUMBER OF CORRECT CLASSES	NUMBER OF INCORRECT CLASSES	% CORRECT CLASSES
Test #2 (33 Documents)	141	100	60%
Text #5 (33 Documents)	132	115	52%

TABLE 9.1 Compound Document Experiment -- Choosing a Primary Class When Found

Another experiment was conducted on the compound documents which utilized the information that a primary class had been selected previously. For final selection as a class, that class must have satisfied the criteria at least twice. The results of this experiment are shown in Table 9.2, and are most encouraging.

COMPOUND DOCUMENT SET	NUMBER OF CORRECT CLASSES	NUMBER OF INCORRECT CLASSES	% CORRECT CLASSES
Test #2 (33 Documents)	78	17	82%
Test #5 (33 Documents)	81	21	79%

TABLE 9.2 Compound Document Experiment -- Requiring a Class to Satisfy the Criteria Twice

9.5 Conclusions

Because no actual Intelligence Report documents were available on which to test this software, these experiments were terminated, even though several other ideas for further improvement were not yet explored. Since the number of incorrect classes had been decreased to an acceptable level by requiring that a class pass the criteria multiple times, the next approach would be to increase the number of correct classes without appreciably increasing the number of incorrect classes.

The multiplicity of a class passing the criteria is a good approach, but needs to be investigated more systematically. How many times should this occur, and upon what factors will this depend for best performance?

It is known that there are sections of reports where no real subject is discussed. The Bayesian distance should be able to detect this, and not attempt to choose any class during this portion of the document. This brings to mind the concept of a "moving window" of keywords, where if no real progress is detected at the front end of the document, then keywords can be dropped off the other end.

This same idea of a "moving window" could be applied more generally even for portions of the document where classification decisions can be made. It is unlikely that if the subject is apt to change that keywords read some time ago will still be useful in determining a class for the portion of the document presently being read. The "moving window" might be applied to drop out such words from memory.

The techniques described in this chapter were not implemented. If Intelligence Reports or other reports which tend to change subject prove to be a problem for the CIRC II classification system, this approach should be considered for implementation.

CHAPTER 10

FINAL EXPERIMENTS WITH THE CIRC II CLASSES AND CONCLUSIONS

10.1 Final Experiments

A comprehensive set of documents sufficiently representative of the entire CIRC II Data Base was not available to thoroughly test the sequential classification software and the final 98 CIRC II classes. Yet a good set of keywords had to be selected, appropriate parameters chosen for the operation of the sequential algorithm, and computer timing verifications made. For these purposes, several sets of test documents were chosen. Two sets of available CIRC II documents were selected for evaluation, consisting of 100 and 104 documents. These two sets shall be referred to as Test Sets #8 and #9, respectively. Another set of 250 documents in IPIR format were used as a final execution and timing test of the BAL software, which will be referenced as Test Set #10.

10.2 Keyword Selection

Keyword selection for the final software was based upon the studies made in Section 3.4. The same notation will be used to describe keyword selection criteria as defined at that point. In addition, let $f(W_i|C_j)$ denote the frequency count for word W_i in class C_j, and $TOTFREQ(C_j)$ the sum of these C_j counts over all keywords. Then for word W_i form the ratio

$$R_{ij} = \frac{f(W_i|C_j)}{TOTFREQ(C_j)} \tag{10.1}$$

for each class C_j. Next normalize these ratios as :

$$NR_{ij} = \frac{R_{ij}}{\sum_{i=1}^{m} R_{ij}} \quad . \tag{10.2}$$

Then let (TOPk) represent the sum of largest k normalized ratios NR_{ij} for word W_i. It is clear that $TOTFREQ(C_j)$ data must be available for the (TOPk) keyword criteria. This will be obtained from a set of keywords selected by a set of simpler criteria which will always contain the final keyword set.

All selection criteria began with the 52,761 distinct non-stoplist words identified by the KEYFINDER software. Tables 10.1 and 10.2 indicate the keyword set criteria evaluated. The criteria in Table 10.1 involve only simple keywords, where the objective is to exclude both low frequency words and high frequency words as discussed in Chapter 3. Words whose total

KW SET	NUMBER OF WORDS	SELECTION CRITERIA	WORDS REMOVED
KSET 1	5,129	$F \geq 20$	45,656
		$F/CT \geq 2.5$ and $\frac{(F - C1 - 2C2)}{(CT - C1 - C2)} \geq 4.5$	1,707
KSET 2	4,431	$F \geq 20$	45,656
		$\frac{(F - C1 - 2C2)}{(CT - C1 - C2)} \geq 5.0$ and any one of these: $F < 80$; TOP 5 $\geq$ 20%; TOP 10 $\geq$ 35%; TOP 22 $\geq$ 55%	2,674
KSET 3	4,178	$F \geq 20$	45,656
		$\frac{(F - C1 - 2C2)}{(CT - C1 - C2)} \geq 5.0$ and any one of these: $F < 80$; TOP 5 $\geq$ 30%; TOP 10 $\geq$ 45%; TOP 22 $\geq$ 67%	2,927

TABLE 10.1 Keyword Set Selection for the 87 CIRC II Classes - Without Compound Keywords, Frequency Table Includes 1's and 2's

KW SET	NUMBER OF WORDS	SELECTION CRITERIA
CSET 3	4178 (Plus 924 CKW)	Same as KSET 3, but CKW added; Frequency Table Does Not Include 1's and 2's
CSET 4	4003 (Plus 924 CKW)	Same as CSET 3, But 178 Word Throw List Added; Does Not Include 1's and 2's
ASET 4	4003 (Plus 924 CKW)	Same as CSET 4, But Frequency Table Includes 1's and 2's
ASET 5	4145 (Plus 467 CKW)	Same as KSET 3, Reduced CKW Set, 425 Word Throw List, 384 Word Keep List, Includes 1's and 2's
ASET 6	4145 (Plus 467 CKW)	Same as ASET 5, But Normalized

TABLE 10.2 Keyword Set Selection for the 87 CIRC II Classes - Including Compound Keywords

frequency counts are less than 20 are discarded as they did not occur sufficiently often in the sample documents to indicate their usefulness as keywords. A number of criteria to reject high frequency words were studied, but the most effective were of the type TOP k $\geq$ constant. It was known from a storage point of view that only about 4,000 simple keywords could be stored and processed by the sequential classification software, so a monotonic decrease in the number of keywords is achieved in KSET 1, KSET 2, and KSET 3. More importantly, the criteria were operating in such a way to reject words with high frequency unsuitable for keywords. KSET 3 represents the best automatic keyword selection criterion studied.

Table 10.2 shows how a number of additional keyword sets can be produced from KSET 3, especially by adding compound keywords. CSET 3, for example, is formed by just taking the union of KSET 3 and the 924 compound keywords initially input to KEYFINDER. CSET 4 is formed from CSET 3 by deleting words from a 178 word throw list. This throw list was formed by noting high frequency words which are not appropriate keywords, but could not be rejected by any automatic criteria. In the classification experiments reported in the next section, it was found that the way the frequency tables were stored could cause certain problems. Specifically, in order to save stoage, frequency counts of ones and twos had been deleted when compound keywords had been added in CSET 3 and CSET 4. When these counts were restored for all keywords, the overall set had improved classification properties, and this was done for sets ASET 4, ASET 5, and ASET 6.

A study was made of the compound keywords, and those compound keywords deleted which had zero or very low frequency counts. Some were retained with low frequency counts if they were deemed sufficiently important for certain classes, or were associated with classes not yet defined. In this process, the number of compound keywords was reduced from 924 initially to a final count of 467. In addition for ASET 5, the throw list and keep list was expanded to yield a final count of 4145 simple keywords and 467 compound keywords. Set ASET 6 is the same as ASET 5, except the normalization process described in Section 7.3 has been applied.

In the next section, classification experiments are conducted on Test Sets #8 and #9 using these keyword sets.

10.3 Evaluation of CIRC II Classification

The keyword sets defined in Tables 10.1 and 10.2 were evaluated by utilizing them to classify document Test Sets #8 and #9. Since these were in CIRC II output format, they were processed by the PL/I software version of CLASSIFY. This also produced maximum output, including keywords examined from each document, so that as much information as possible can be gained about the keyword set investigated.

It was not possible here to obtain a final keyword set with superior classification properties. Instead, a trend will be indicated to show that such superior performance is achievable, and how this can be accomplished.

FTD will have to continue these keyword set improvements in order to achieve the best possible results, but this can only be done after eleven more classes are defined and a comprehensive set of test documents selected for evaluation. The experiments in Section 3.4 for UDC classes clearly indicated that the best performance that can be achieved is about 80% of the classes chosen correct and 90% of the documents assigned at least one correct class. A number of trade offs are encountered which prevent much better accuracy than this from being achieved. For example, if one tries to obtain at least one correct class for nearly every classifiable document, then the % correct classes criterion will almost certainly suffer, and decrease. Also, more keywords will have to be read to achieve improved classification accuracy, until the entire document has to be read, and this will require increased processing time. Another trade off is that more and more specific keywords will have to added to achieve this increase in performance, adding an unacceptable increase in core storage, or add an unacceptable processing time increase to access keyword data in peripheral storage. The experiments conducted in this study will illustrate some of these trade offs.

Table 10.3 summarizes the results of these experiments. Notice that the classification performance is better with Test Set #9 than Test Set #8. This is because Test Set #8 contains several report documents which are more difficult to classify than strictly technical abstracts dealing with one specific technical area.

KSET 3 utilizes only 4,178 simple keywords, whereas all other keyword sets in TABLE 10.3 also contains compound keywords. Notice how the performance deteriorates with keyword set CSET 3. This is partially due to the fact that 1's and 2's have been deleted from the frequency table for CSET 3. Also both KSET 3 and CSET 3 contain inappropriate high frequency keywords not rejected by the automatic keyword criteria. This leads to somewhat inconsistent results when 1's and 2's have been dropped from the keyword tables, and classes are dropped too rapidly. In order to correct this, the default parameter δ is increased to 5×10^{-5}. Recall that as δ is increased, classes are retained longer and more keywords tend to be read before a classification decision is made. As a result of the change in δ, there is a dramatic improvement for both Test Sets #8 and #9.

In CSET 4, 178 of the inappropriate keywords were manually removed by placing them on the throw list, and CSET 4 now consists of only 4003 simple keywords plus 924 compound keywords. Table 10.3 shows that when CSET 4 was used with $\delta = 5 \times 10^{-5}$, the classification results again improved dramatically for both Test Sets #8 and #9. The improvement is in both the % classes correct and document accuracy criteria. However, a price has been paid for this improvement, for with a decreased keyword set, now some documents no longer contain a sufficient number of keywords, and are identified as being unclassifiable (UNCL) since they contain fewer than T = 6 keywords. When unclassified documents are reported as in this case, it should be observed that the document accuracy criterion reports the percent of all documents which have at least one assigned class correct. If the number of unclassifiable documents were removed from consideration, the percentages in Table 10.3 for document accuracy would be even higher. It might also be noted

KW SET T, δ	DOC. SET	NUMBER OF CORRECT CLASSES	NUMBER OF INCORRECT CLASSES	% CORRECT CLASSES	DOCUMENT ACCURACY
KSET 3	#8	85	72	54%	68%
6 5×10^{-6}	#9	99	55	64%	80%
CSET 3	#8	70	78	47%	61%
6 5×10^{-6}	#9	86	57	60%	70%
CSET 3	#8	81	67	55%	69%
6 5×10^{-5}	#9	100	60	63%	81%
CSET 4	#8	88	62	59%	(2 UNCL)[a] 73%
6 5×10^{-5}	#9	102	49	68%	(7 UNCL) 80%
ASET 4	#8	94	52	64%	(1 UNCL) 74%
6 5×10^{-6}	#9	102	46	69%	(7 UNCL) 79%
ASET 5	#8	98	62	61%	77%
5 5×10^{-6}	#9	101	45	69%	(6 UNCL) 78%
ASET 5	#8	106	57	65%	(1 UNCL) 83%
5 5×10^{-5}	#9	110	47	70%	(7 UNCL) 84%
ASET 6	#8	107	56	66%	(1 UNCL) 84%
(Normalized)	#9	109	41	73%	(7 UNCL) 84%
5 5×10^{-5}					

[a] UNCL Means Document is Unclassifiable

TABLE 10.3 Classification Results for the 87 CIRC II Classes

that the reason why δ was chosen at an increased level is to tend to counteract the fact that 1's and 2's are still deleted from the keyword frequency table.

ASET 4 reinstates the 1's and 2's in the frequency table and δ is decreased correspondingly for the experiment. Improved classification is achieved, especially for Test Set #8, where improvement is really needed. Only marginal improvement is observed in Test Set #9.

In the next experiments with ASET 5, a number of changes were implemented. If more time had been available, a more careful and systematic set of experiments would have been conducted to change only one variable or parameter at a time. First, the compound keyword frequencies were studied, and a decision was made to delete about half of them, as their frequencies were far too low. A set of 467 compound keywords were retained which either possessed respectable frequency counts, were deemed essential for certain classes with few specific keywords, or corresponded to classes for which no defining documents had yet been analyzed by KEYFINDER. The second major change was that the throw list was increased to 425 words in order to eliminate inappropriate high frequency words, and 384 words were added to the keep list which had previously been rejected by the automatic keyword criteria. A third change was to reduce the minimum number of keywords required to classify a document to T = 5, in order to reduce the number of documents declared unclassifiable. Experiments were conducted for these conditions with ASET 5 using $\delta = 5 \times 10^{-6}$ and 5×10^{-5}. For the smaller default value, there is at most only a marginal improvement for all these changes, but the larger default value yields a greater benefit of these modifications, as both Test Sets #8 and #9 improve considerably in both % classes correct and document accuracy. Notice, however, that no real change has occurred in the number of unclassifiable documents -- they still contain too few keywords.

A final experiment was made with keyword set ASET 5 normalized as described in Section 7.3, and this is termed ASET 6. With an increased default parameter δ, there is another small improvement in both Test Sets #8 and #9.

These classification results are still not optimal, but this set of experiments have shown that sustained improvement in classification accuracy can be achieved by keyword selection techniques, and these accuracy figures are definitely approaching the 80% correct classes and over 90% document accuracy objectives. For example, for Test Set #9, if the seven unclassifiable documents were removed from this 104 document set, and only incorrectly classified documents targeted, a modified document accuracy figure of 90% has already been achieved.

These classification results can be improved by:

1) better sample documents selected to define the 98 classes;
2) better frequency counts for compound keywords;
3) optimal selection of keywords through use of the automatic keyword selection criteria, and definition of the throw list and keep list;

4) optimal selection of classification parameters T, R, α, and δ; T and δ seem most effective for this purpose.

10.4 Timing Measures for CIRC II Classification

An effort was made to obtain timing and speed of processing information for the PL/I classification runs conducted in the experiments described in Section 10.3. The problem is that all that was available was the GO-STEP time, which includes CPU processing time, but may include other system time, e.g., WAIT-STATE times for the operating system, I/Ø buffer times, etc. A further problem is that only about 100 documents were classified, and yet a lot of initial preprocessing steps had to be accomplished in order to set up the run. Another fact to consider is that a lot of output was printed for diagnosis in the PL/I version of CLASSIFY which is not done at all with the BAL version, and this requires additional time. Thus the times reported here will be an upper bound on the actual classification times required. Indeed the absolute times are not as informative as are the changes in times as various parameters are modified and different keyword sets utilized.

Table 10.4 shows the running times for the same experiments as reported in Table 10.3. The GO-STEP time is given in seconds for IBM 370/168 computer, and a figure of documents/sec processed on this computer in PL/I at The Ohio State University Instruction and Research Computer Center facility. Comparing the two tables and starting with the CSET 3 experiment, one can generally see that improved accuracy is achieved through the experiment on ASET 6 with an increase in processing time. This illustrates one of the trade offs mentioned in the last section, i.e., if greater accuracy in classification is desired, greater processing time will almost always be required to achieve it. In the next section the processing times reported in Table 10.4 will be related to the IBM 360/65 computer at the FTD facility.

In comparing the KSET 3 experiment to other data in Table 10.4, recall that 1's and 2's were stored in the frequency table for this keyword set, but not for CSET 3 and CSET 4. It is primarily this effect which is seen in the decrease in processing time from KSET 3 to CSET 3, and thus masks the effect of the inclusion of compound keywords. The best comparison can be made between KSET 3 and ASET 4, where it can be seen that the compound keywords and throw list have improved performance appreciably but Table 10.4 shows no significant increase in processing time. The effect of the default δ parameter can clearly be observed in the two CSET 3 runs and two ASET 5 experiments. The performance improved significantly in both cases, but cost approximately a 10% increase in processing time. It may be finally noted that when normalization of the keywords was imposed in ASET 6, the performance improved and the processing time decreased slightly, thus illustrating that improved classification does not always require increased processing time.

KW SET	T,δ	DOCUMENT SET	GO-STEP TIME (SEC)	DOCUMENT PROCESSED PER SECOND
KSET 3	$6_{5x10^{-6}}$	#8 (100 doc)	21.69	4.61
		#9 (104 doc)	21.83	4.76
CSET 3	$6_{5x10^{-6}}$	#8	16.26	6.15
		#9	16.37	6.35
CSET 3	$6_{5x10^{-5}}$	#8	18.13	5.52
		#9	18.25	5.70
CSET 4	$6_{5x10^{-5}}$	#8	17.78	5.62
		#9	17.53	5.93
ASET 4	$6_{5x10^{-6}}$	#8	22.90	4.37
		#9	21.83	4.76
ASET 5	$5_{5x10^{-6}}$	#8	20.33	4.92
		#9	19.73	5.27
ASET 5	$5_{5x10^{-5}}$	#8	22.55	4.43
		#9	22.19	4.69
ASET 6	$5_{5x10^{-5}}$	#8	21.93	4.56
		#9	21.56	4.82

TABLE 10.4 Classification Timings for the 87 CIRC II Classes, IBM 370/168 Computer

10.5 BAL Version of CLASSIFY for Documents in IPIR Format

The basic assembly language (BAL) version of CLASSIFY was run on the 250 Test Set #10 IPIR formatted documents using keyword set ASET 6. A summary of this final testing run is given in Table 10.5 Test Set #10 contains 171 documents without text and is not very appropriate for a final evaluation. Nevertheless the accuracy figures of 71% correct classes and 78% document accuracy are comparable to the results reported in Table 10.3, but it is clear that unclassifiable documents have had to be excluded in both these figures.

The time required to process the 250 documents was 2.98 seconds, for 83.9 documents/sec. Although this includes system setup and preprocessing time, it is probably too optimistic, as far too many of the documents had no text. For example, comparing this to the experiment in Table 10.4 would

KW SET	T,δ	DOCUMENT SET	NUMBER OF CORRECT CLASSES	NUMBER OF INCORRECT CLASSES	% CORRECT CLASSES
ASET 6 (Normalized)	5×10^{-5}	#10 (250 doc)	77	31	71%

DOCUMENTS CORRECT	DOCUMENTS INCORRECT	DOCUMENTS WITH NO TEXT	OTHER DOCUMENTS NOT CLASSIFIED	DOCUMENT ACCURACY
58	16	171	4	78%

TIME (SEC)	DOCUMENTS PROCESSED PER SECOND	CORE STORAGE REQUIRED (BYTES)
2.98 (for 250 doc)	83.9	502 K

TABLE 10.5 Classification Results for BAL CLASSIFY on 250 Documents for 87 CIRC II Classes on the IBM 370/168 Computer

indicate that the BAL program is about 17 times faster than the PL/I software. This is a larger ratio than our past experience would justify, but does indicate how much faster the BAL version of CLASSIFY runs than the PL/I software. The IBM 360/65 computer at the FTD facility has been shown on a number of occasions to execute BAL code about three times slower than the IBM 370/168 computer at Ohio State University. Thus the experiment in Table 10.5 would be expected to run at the FTD facility at the rate of about 28 documents/sec. Previous experiments at the FTD facility for the BAL CLASSIFY software have been executed at about 24 documents/sec.

The core storage required was 502 K bytes for the BAL CLASSIFY software and keyword set ASET 6 with 4145 simple keywords and 467 compound keywords. If core storage were at a premium, this could be reduced immediately by 30 K bytes by more efficiently allocating space for the keyword frequency tables. Further reductions in storage could be accomplished only by reductions in either the simple or compound keywords.

10.6 Conclusions

The final experiments reported in this chapter have shown that the CIRC II classification system can achieve both accurate and rapid classification of CIRC II documents. Although the target figures of 80% correct classes and

90% document accuracy were not obtained, a consistent improvement in that direction was achieved by successively better keyword sets. It is clear that if this keyword selection process were continued, the accuracy objectives could be obtained.

Eleven more CIRC II classes are required for the classification system, and it is urged that the defining sample documents for these classes be carefully selected. As indicated in Chapter 5, the documents which presently define the 87 CIRC II classes should be re-examined, and more documents added to better define some of these classes. This should be a one-time operation, and so is worth a modest investment of time. A careful selection of representative documents at this point will yield better keyword frequencies over classes for use in the classification system. A re-examination of the compound keywords is needed. They appear to definitely improve classification performance, but an excessive number will cause both storage and processing time problems. After all the final sample documents and compound keywords have been selected, it is recommended that the entire KEYFINDER software be rerun to establish the best possible frequency data for both simple and compound keywords.

After these final keyword frequency data are obtained, the only other classification system changes which can affect classification results are keyword selection and final system parameters. The best keywords possible should be selected using the throw list or keep list, and possibly even modifying the automatic keyword criterion in the CONVERT software described in Chapter 7.

The reasons for the recommended final objective figures of 80% correct classes and over 90% document accuracy can be seen in the following trade offs. Classification accuracy can possibly be further improved by the inclusion of more specific keywords, which perhaps occur quite infrequently. However, this may lead to an unacceptable increase in core storage or document processing time, or both. Classification accuracy can possibly be further improved by reading more keywords in each document. This was observed when the default parameter δ was increased. However, this may require too much of the document to be read, and again impose excessive document processing time. An increase in the parameter T can increase the number of keywords read before a decision is made, and thus improve classification accuracy. But then an unacceptable number of documents may be declared unclassifiable which otherwise could usually be correctly classified. The stopping criterion could be modified; for example, if only classes with confidence levels exceeding 0.9 were selected, the experiments in Section 6.4 showed improved accuracy might be obtained. But then at most one class would be chosen for each document classified, and an unacceptable number of documents would be declared unclassifiable.

It is hoped that this CIRC II classification system will be implemented, and will serve as a viable solution to the CIRC II Data Base problems identified in Chapter 1.

REFERENCES

1. Aberi, H., "Implementation of the Fried Model of Automatic Classification", M.S. Thesis, The Ohio State University, Columbus, Ohio (1970).

2. Brinkman, B. J., "KEYFINDER System for the OSU Sequential Classifier", Computer Program Documentation for Contract F30602-76-C-0102, Department of Computer and Information Science, The Ohio State University, March 31, 1977.

3. Crawford, L. G., "CLASSIFY System for the OSU Sequential Classifier", Computer Program Documentation for Contract F30602-76-C-0102, Department of Computer and Information Science, The Ohio State University, March 31, 1977.

4. Day, A., "Full Table Quadratic Searching for Scatter Storage", Comm. ACM, Vol. 13, #8, 1970, pp. 481-482.

5. Fried, J. B. and Landry, B. C., et al., "Index Simulation Feasibility and Automatic Document Classification", Technical Report 68-4, NSF Grant GN-534, Computer and Information Science Research Center, The Ohio State University, 1968.

6. Fu, K. S., Sequential Methods in Pattern Recognition and Machine Learning, Academic Press, New York, 1968.

7. Kar, B. G. and White, L. J., "A Distance Measure for Automatic Sequential Document Classification", Technical Report 75-7, NSF Grant GN-36340, Computer and Information Science Research Center, The Ohio State University, Columbus, Ohio, 1975.

8. Salton, G., "A Theory of Term Importance in Automatic Indexing", Journal of the ASIS, Vol. 26, #1, January-February 1975, pp. 33-44.

9. Universal Decimal Classficiation, Abridged English Edition, B.S. 1000A: 1961, 3rd Edition, British Standards Institution, London, 1961.

10. Wald, A., Sequential Analysis, Chapter 10, John Wiley and Sons, Inc., New York, 1947.

11. White, L. J., Smith, J. D., Kar, G., Westbrook, D. E., Brinkman, B. J., Fisher, R. A., "A Sequential Method for Automatic Document Classification", Technical Report 75-5, NSF Grant GN-36340, Computer and Information Science Research Center, The Ohio State University, Columbus, Ohio, 1975.

APPENDIX A

COSATI CLASS FREQUENCIES

Statistics were taken from documents disseminated January - May 1976.

	DOCUMENTS	COSATIs
File A	4,744	6,297
File B	3,316	4,595
File C	134,580	166,897
TOTAL	142,670	177,789

COSATI	FILE A	FILE B	FILE C	TOTAL %
00	.5	2	0	2.00
01	7	2	13	2.75
02	1	3	.5	2.50
03	.5	1	.5	.75
04	1	1	.5	1.00
05	7	6	5	6.50
06	15.5	18	.5	17.25
07	2	9	.5	8.50
08	2	7	.5	6.50
09	6	6	7	5.75
10	2	1	.5	1.00
11	3	7	.5	6.75
12	.5	3	.5	2.25
13	11	14	5	13.50
14	3	3	.5	3.50
15	6	1	7	1.50
16	6	.5	22	1.25
17	9	2	19	2.75
18	3	1	3	1.25
19	5	.5	1	.75
20	4	10	1	9.75
21	2	1	1	1.00
22	3	1	11	1.25
TOTAL CODES	4595	166,897	6297	177,789

APPENDIX B

110 UDC CLASSES

CLASS	DESCRIPTION	UDC CODE RANGE	ESTIMATED % OF DOCUMENTS
1	Generalities	0	0.71
2	Philosophy, Psychology, Ethics, Religion, Theology	1/2	0.32
3	Social Sciences, Economics	3	1.12
4	Science in General and Mathematics (Excluding Calculus & Probability)	5 only 50 only 51 only 510/516 518	0.76
5	Calculus	517	1.06
6	Probability	519	0.60
7	Astronomy	52 only 520/524	0.57
8	Earth, Surveying, Geology, Navigation, Chronology	525/529	0.91
9	Physics and Mechanics General Principles	53 only 530/531	1.26
10	Fluid Mechanics	532	0.71
11	Gas Mechanics	533	0.54
12	Vibration and Acoustics	534	0.61
13	Optics and Light	535	1.17
14	Heat and Thermodynamics	536	0.98
15	Electricity	537	0.73
16	Magnetism	538	0.59

17	Physical Nature of Matter	539 only 539.0, 539.2/.9	together 2.43
18	Nuclear Physics	539.1	
19	Chemistry and Minerology	54 only, 540	together 3.20
20	General Theoretical and Physical Chemistry; General Chemistry	541 only, 541.0 541.3/.9	
21	Physical Chemistry	541.1	
22	Atomic Theory (isotopes)	541.2	
23	Experimental Chemistry	542	0.56
24	Analytical Chemistry and Quantitative Analysis	543/545	1.41
25	Inorganic Chemistry	546	1.29
26	Organic Chemistry - Acyclic Compounds Organic Chemistry - Natural Substances of unknown composition	547 only 547.0/,4 547.9	together 2.09
27	Organic Chemistry - Cyclic Compounds	547.5/.8	
28	Crystallography and Minerology	548/549	1.06
29	Geology in General Geochemistry, Geobiology, Applied Geology	55 only 550 only 550.0/.2 550.4/.9	together 3.48
30	Geophysics/(earthquakes) Form, Structure, Origin of the Earth, Geodynamics (volcanoes) Physical Geography, Topography	550.3 551 only 551.0/.4	
31	Meterology and Climatology	551.5/.6	
32	Historical Geology, Stratigraphy, Paleogeography Paleontology, Fossils	551.7/.9 56	

33	Petrology	552	0.39
34	Economic Geology, Ores, Minerals, Deposits, Exploration	553/559	0.65
35	Anthropology, Biology, Archeology, Prehistoric Man	57 only 570/574 577/579	0.85
	General Properties of Life		
36	Genetics, Development of Organisms, Evolution, Origin of Life	575/576	0.96
37	Botany	58	1.19
38	Zoology	59	1.11
39	Medical Sciences Anatomy, Comparative Pathology	61 only 610/611 617	0.87
	Surgery, Orthopaedics	619	
	Comparative Pathology, Veterinary Medicine		
40	Physiology	612	1.17
41	Health, Preventive Medicine, Public Health and Safety	613/614	0.95
42	Toxicology, Pharmacology	615	1.35
43	Disease, Pathology, and Medicine	616 only 616.0	together 3.76
	Diseases: Respiratory, Digestive, Glands, Skin, Urology, Skeletal System	616.2/.7 618	
	Gynecology, Obstetrics		
44	Circulatory, Cardiovascular, and Blood Disease	616.1	
45	Neurology and Psychiatry	616.8	
46	Infectious, Communicable Diseases	616.9	

47	Engineering and Technology Generally	6 only 60 only 62 only	1.42
	General History of General Technology, Inventions and Patents		
48	Materials Testing	620 only 620.0/.3	together 1.73
49	Power Stations, General Economics of Energy	620.4/.9	
50	Mechanical and Electrical Engineering in General, Machinery in General, Mechanical Engineering Theory and Principles	621 only 621.0	0.95
51	Steam Power Engines, Boilers, Water Power, Hydraulic Energy	621.1/.2	0.98
52	Electrical Engineering Generally	621.3 only 621.30	0.89
	Electrical Lighting, Lamps	621.32	
53	Power Supply, Distribution, and Control	621.31 only 621.310 621.317/.319	
	Measurements, Instruments, Indicators, Applied Magnetism and Electrostatics		together 5.20
54	Power Generation, Power Stations, Electrical Networks	621.311	
55	Production of Electrical Accessories, Electrical Manufacturing Industry	621.312 621.314/.316	
	Transformers, Transmission Lines, Wires, Switches, Relays, Fuses		
56	Motors, Generators	621.313 621.33/.34	
	Electric Traction and Electric Drives		
57	Electrochemistry, Thermoelectricity, Electric Heating	621.35/.36	0.50

AD-A042 268 OHIO STATE UNIV COLUMBUS DEPT OF COMPUTER AND INFORM--ETC F/G 5/2
CIRC II DATA BASE CLASSIFICATION.(U)
JUN 77 L J WHITE, A E PETRARCA, L G CRAWFORD F30602-76-C-0102

UNCLASSIFIED RADC-TR-77-211 NL

2 OF 2
ADA042268



END

DATE
FILMED
8-77

58	Technique of Electric and Electromagnetic Waves, Oscillations, and Pulses, Radiation, Guided Propagation, Electric Generators and Oscillators	621.37 only 621.370/.374 621.377/.379	together 1.84
59	Amplifiers, Modulators, and Detectors	621.375/.376	
60	Electronics	621.38	1.03
61	Telecommunication Telegraphy, Telephony	621.39 only 621.390/.395	together 1.49
62	Radiocommunications, Radio Transmitters, Receivers, Radar, Television	621.396/.399	
63	Internal Combustion and Other Engines	621.4	0.75
64	Pneumatic Energy Refrigeration, Heat Pumps	621.5	0.53
65	Fluid Distribution, Storage Containers, Pipes, Pumps	621.6	0.75
66	Workshop Practice, Fabrication Powder Metallurgy Metallization, Chemical Finishing, Warehouses, Depots, Packing, Dispatch	621.7 only 621.70 621.76 621.793/.799	together with 69, 1.98
67	Pattern and Die Making, Forges and Forging, Foundaries, Tool Making	621.71/.75	1.11
68	Rolling, Drawing, Boiler-Making, Sheets, Tubes, Pipes	621.77/.78	1.26
69	Welding, Soldering	621.79 only 621.790/.792	together with 66, 1.98

70	Power Transmission, Materials Handling, Mechanical Fixing, Attachment, Lubrication	621.8 only 621.80/.81 621.86/.89	together 1.84
	Materials Handling, Hoisting, Cranes, Jacks, Lubrication		
71	Transmissions, Bearings, Bushings, Gears, Cams, Clutches, Links, Linkages, Pulleys, Wheels, Chains	621.82/.85	
72	Tools, Machine Tools, Machinery Planing, Milling, Grinding, Polishing	621.9 only 621.90/.92 621.97/.99	together 1.89
	Perforating, Shearing, Presses Screw Cutting		
73	Saws, Lathes, Drills, Punches	621.93/.96	
74	Mining and Mineral Dressing Exploration, Sampling, and Analysis Specific Minerals, ore, coal, oil fields, mine services, mine safety	622 only 622.0/.1 622.3/.5 622.8/.9	0.75
75	Mining Operations	622.2 only 622.20/.22	together 1.79
	Methods of Mine Working, Supports	622.26/.29	
76	Excavation, Boring, Drilling	622.23/.25	
77	Haulage and Handling, Mineral Dressing, Ore Preparation	622.6/.7	0.87
78	Military Engineering	623 only 623.0/.7	together 1.56
	Civil and Structural Engineering	624	
79	Naval Engineering	623.8/.9 626/627	
	Hydraulic Engineering, River, Port, Harbor, and Coast Works, Dams		
80	Railway, Highway Engineering	625	0.80
81	Public Health Engineering	628	0.42
82	Transport Engineering	629	1.28

83	Agriculture, Gardens, Gardening Fruit Cultivation, Horticulture, Insect and Reptile Breeding and Management, Game and Fish Management	63 only 630 634/635 638/639	1.16
84	Agronomy, Farming Generally Soil Science Rural Engineering Agricultural Influences, Ecology	631 only 631.0/.2 631.4 631.6/.7 631.9	together 1.71
85	Farm Operations, Growing, Cultivation Fertilizers, Manuring	631.5 631.8	
86	Plant Diseases, Pests, Crop Damage, Field Crops	632/633	1.16
87	Stockbreeding, Livestock, Domestic Animals, Pets, Dairy Milk Products	636/637	0.57
88	Domestic Science, The Home, Commerce, Office, Business Management, Publicity, Advertising	64,65 only 650/656 659	together 1.86
89	Accounting, Bookkeeping, Business, Factory Management	657/658	
90	Metallurgy, Chemical Engineering	66 only 660	1.58
91	Chemicals (Fine, Heavy)	661	0.82
92	Explosives, Fuels	662	0.59
93	Beverages, Stimulants, Food Industry	663/664	0.83
94	Oils, Fats, Waxes	665	0.56

95	Glasses and Ceramics	666 only 666.0	
	Ceramics and Clay industry, Cement, Concrete	666.3/.9	together 1.74
96	Glass Industry	666.1/.2	
97	Dyes, Paints, Organic Chemicals	667/668	0.37
98	Metallurgy	669 only 669.0	
	Other Non-Ferrous Metals	669.3/.9	
99	Ferrous Metals, Iron and Steel	669.1 669.24/.29	together 3.31
	Metals for Alloy Steels		
100	Precious Metals and Their Alloys, Gold, Silver	669.2 669.20/.23 671	
	Precious Metal, Gem Industries, Jewelry		
101	Industries and Crafts Based on Processable Materials	67 only 670 672/673	
	Iron and Steel Goods, Non-ferrous Metal Goods	675 679	
	Leather Industry		
	Other Industries, Stones, Minerals		together 3.86
102	Timber and Wood Industry Paper and Pulp Industries	674 676	
103	Textiles and Fibers	677	
104	Rubbers and Plastics	678	
105	Crafts and Special Trades for Finished Articles and Goods	68 only 680 682/686	
	Ironwork, Hardware, Furniture Books, Office Materials	688/689	together 2.75
	Fancy and Decorative Goods, Hobbies and Handicrafts		
106	Instruments and Machines	681	

107	Clothing, Brushes, Toilet Industry	687	
108	Construction Industry	69	1.17
109	Arts, Recreation, Entertainment, Sports Principally Photography, Cinema, Architecture	7	0.54
110	Literature, Geography, History, Biography (also Language and Linguistics)	8/9 4	0.52

APPENDIX C
UDC CLASS FREQUENCY DATA

TABLE C1

DISTRIBUTION OF DOCUMENTS
BY UDC AT ONE DIGIT ROOT

SUBJECT AREA	UDC ROOT	# OF DOCUMENTS	PERCENT OF TOTAL
Generalities	0	1488	0.7
Ethics, Philosophy, Psychology	1	419	0.2
Theology	2	245	0.1
Social Security, Economics	3	2346	1.1
Linguistics, Languages	4	185	0.1
Math and Natural Sciences	5	64866	31.1
Applied Science, Medicine, Technology	6	137237	65.7
Recreation and Sports	7	1137	0.5
Literature	8	179	0.1
Geography and History	9	714	0.3
		208815	99.9

TABLE C2

DISTRIBUTION OF DOCUMENTS
BY UDC AT TWO DIGIT ROOT

UDC ROOT	# OF DOCUMENTS	PERCENT OF TOTAL	UDC ROOT	# OF DOCUMENTS	PERCENT OF TOTAL
00	716	0.34	41	17	0.01
01	227	0.11	42	20	0.01
02	68	0.03	43	31	0.01
03	14	0.01	44	15	0.01
04	22	0.01	45	17	0.01
05	22	0.01	46	11	0.01
06	348	0.17	47	33	0.02
07	18	0.01	48	18	0.01
08	32	0.01	49	19	0.01
09	21	0.01	50	46	0.02
10	32	0.02	51	5010	2.40
11	22	0.01	52	3075	1.47
12	69	0.03	53	18827	9.02
13	25	0.01	54	20081	9.62
14	37	0.02	55	9145	4.38
15	61	0.03	56	294	.14
16	51	0.02	57	3786	1.81
17	34	0.02	58	2492	1.19
18	36	0.02	59	2110	1.01
19	52	0.02	60	102	0.05
20	5	----	61	16642	7.97
21	26	0.01	62	70137	33.59
22	20	0.01	63	9608	4.60
23	23	0.01	64	123	0.06
24	36	0.02	65	3892	1.86
25	18	0.01	66	20472	9.80
26	34	0.02	67	8052	3.86
27	25	0.01	68	5749	2.75
28	33	0.02	69	2460	1.17
29	25	0.01	70	5	----
30	60	0.03	71	81	0.04
31	116	0.06	72	217	0.10
32	40	0.02	73	19	0.01
33	802	0.38	74	99	0.05
34	123	0.06	75	14	0.01
35	238	0.11	76	20	0.01
36	71	0.03	77	623	0.30
37	237	0.11	78	30	0.01
38	631	0.30	79	29	0.01
39	28	0.01	80	9	----
40	4	----	81	10	----

TABLE C2

UDC ROOT	# OF DOCUMENTS	PERCENT OF TOTAL
82	26	0.01
83	28	0.01
84	18	0.01
85	15	0.01
86	25	0.01
87	10	----
88	19	0.01
89	19	0.01
90	14	0.01
91	457	0.22
92	123	0.06
93	26	0.01
94	32	0.02
95	26	0.01
96	5	----
97	11	0.01
98	11	0.01
99	9	----

TABLE C3

DISTRIBUTION OF DOCUMENTS AT THREE DIGIT ROOT FOR CATEGORIES 5 and 6

UDC ROOT	# OF DOCUMENTS	PERCENT OF TOTAL	UDC ROOT	# OF DOCUMENTS	PERCENT OF TOTAL
51 only	220	0.11	548	1502	0.72
510	2	----	549	714	0.34
511	123	0.06	55 only	177	0.08
512	138	0.07	550	2958	1.42
513	480	0.23	551	3857	1.85
514	7	----	552	804	0.39
515	84	0.04	553	1107	0.53
516	17	0.01	554	5	----
517	2217	1.06	555	14	0.01
518	479	0.23	556	189	0.09
519	1243	0.60	557	16	0.01
52 only	77	0.04	558	---	----
520	4	----	559	18	0.01
521	101	0.05	56 only	86	0.04
522	78	0.04	560	---	----
523	916	0.44	561	59	0.03
524	9	----	562	15	0.01
525	94	0.05	563	36	0.02
526	5	----	564	40	0.02
527	14	0.01	565	14	0.01
528	1748	0.84	566	6	----
529	29	0.01	567	13	0.01
53 only	328	0.16	568	9	----
530	299	0.14	569	16	0.01
531	1997	0.96	57 only	77	0.04
532	1487	0.71	570	5	----
533	1122	0.54	571	15	0.01
534	1278	0.61	572	36	0.02
535	2435	1.17	573	4	----
536	2038	0.98	574	37	0.02
537	1530	0.73	575	330	0.16
538	1238	0.59	576	1678	0.80
539	5075	2.43	577	1461	0.70
54 only	260	0.12	578	138	0.07
540	18	0.01	579	5	----
541	6405	3.07	58 only	59	0.03
542	1162	0.56	580	13	0.01
543	2868	1.37	581	1794	0.86
544	11	0.01	582	607	0.29
545	71	0.03	583	3	----
546	2699	1.29	584	6	----
547	4371	2.09	585	3	----

TABLE C3

UDC ROOT	# OF DOCUMENTS	PERCENT OF TOTAL	UDC ROOT	# OF DOCUMENTS	PERCENT OF TOTAL
586	1	----	638	43	0.02
587	4	----	639	438	0.21
588	2	----	66 only	3300	1.58
589	---	----	660	32	0.02
59 only	59	0.03	661	1703	0.82
590	---	----	662	1227	0.59
591	813	0.39	663	537	0.26
592	20	0.01	664	1186	0.57
593	62	0.03	665	1160	0.56
594	25	0.01	666	3637	1.74
595	663	0.32	667	472	0.23
596	12	0.01	668	301	0.14
597	133	0.06	669	6917	3.31
598	113	0.05	69 only	1084	0.52
599	210	0.10	690	7	----
61 only	131	0.06	691	665	0.32
610	9	----	692	8	----
611	338	0.16	693	245	0.12
612	2443	1.17	694	7	----
613	984	0.47	695	3	----
614	1001	0.48	696	52	0.02
615	2825	1.35	697	271	0.13
616	7620	3.65	698	18	0.01
617	869	0.42	699	100	0.05
618	237	0.11			
619	532	0.25			
62 only	2966	1.42			
620	3605	1.73			
621	47978	22.98			
622	7116	3.41			
623	133	0.06			
624	2212	1.06			
625	1665	0.80			
626	471	0.23			
627	447	0.21			
628	875	0.42			
629	2669	1.28			
63 only	107	0.05			
630	61	0.03			
631	3580	1.71			
632	1667	0.80			
633	761	0.36			
634	1588	0.76			
635	169	0.08			
636	657	0.31			
637	537	0.26			

TABLE C4

FOUR DIGIT ROOT DISTRIBUTION FOR UDC 621,622

UDC ROOT	# OF DOCUMENTS	PERCENT OF TOTAL
	C4	
621 only	687	0.33
621.0	1301	0.62
621.1	1608	0.77
621.2	435	0.21
621.3	22839	10.94
621.4	1565	0.75
621.5	1106	0.53
621.6	1561	0.75
621.7	9087	4.35
621.8	3837	1.84
621.9	3952	1.89
622 only	136	0.07
622.0	100	0.05
622.1	87	0.04
622.2	3744	1.79
622.3	731	0.35
622.4	150	0.07
622.5	45	0.02
622.6	878	0.42
622.7	938	0.45
622.8	287	0.14
622.9	20	0.01

TABLE C5

FIVE DIGIT ROOT DISTRIBUTION FOR FOR UDC 621.3, 621.7

UDC ROOT	# OF DOCUMENTS	PERCENT OF TOTAL
	C5	
621.3 only	1626	0.78
621.30	15	0.01
621.31	10311	4.94
621.32	213	0.10
621.33	454	0.22
621.34	80	0.04
621.35	744	0.36
621.36	283	0.14
621.37	3850	1.84
621.38	2155	1.03
621.39	3108	1.49
621.7 only	238	0.11
621.70	9	----
621.71	13	0.01
621.72	31	0.01
621.73	429	0.21
621.74	1706	0.82
621.75	152	0.07
621.76	319	0.15
621.77	1692	0.81
621.78	940	0.45
621.79	3558	1.70

APPENDIX D

A SAMPLING OF KEYWORDS CHOSEN BY CLASS

LIST OF WORDS IN CLASS 36 - MOTORS

WORD	COUNT	WORD	COUNT	WORD	COUNT
ARMATURE	42	MOTOR	191	SLIP	11
ASYNCHRONOUS	11	MOTORS	84	STATOR	72
BRUSH	27	POLE	15	SYNCHRONOUS	55
COIL	51	POLES	9	TACHOGENERATOR	10
COILS	31	REACTANCE	11	TRACTION	17
EXCITATION	87	RECTIFIER	36	WINDING	103
EXCITER	18	ROTOR	89	WINDINGS	39
INDUCTIVE	14	ROTORS	17		

LIST OF WORDS IN CLASS 37 - BATTERY

WORD	COUNT	WORD	COUNT	WORD	COUNT
ANODE	62	CATHODE	68	ELECTROLYTIC	25
ANODES	16	CATHODES	6	ELECTROLYZER	16
BATH	45	ELECTRCHEMICA	15	PALLADIUM	9
BATTERY	45	ELECTRODES	77	POLARITY	17
BATTERIES	60	ELECTROLYSIS	9	THERMIONIC	7
CADMIUM	39	ELECTROLYTE	121		

LIST OF WORDS IN CLASS 38 - FURNACES

WORD	COUNT	WORD	COUNT	WORD	COUNT
ASH	29	FLUE	13	KILNS	15
BLAST	24	FLUIDIZED	12	MELTING	25
BOILER	28	FURNACE	209	OVEN	12
BOILERS	16	FURNACES	56	ROASTING	18
BURNER	35	GASES	36	SINTERING	12
BURNERS	22	HEARTH	8	SLAG	36
CALCINATION	7	HEATING	99	SMELTING	33
CHARGE	37	KILN	85		

LIST OF WORDS IN CLASS 39 - OIL/LUB

WORD	COUNT	WORD	COUNT	WORD	COUNT
ADDITIVES	62	GREASE	11	LUBRICATION	29
ANTICORROSION	26	HYDRODYNAMIC	22	OILS	77
AUTOMOTIVE	7	HYDROSTATIC	13	PARAFFIN	8
COLLOIDAL	7	INCLUSIONS	10	SLIPPING	6
FLASH	25	LUBRICANT	132	VISCOSITY	106
FOAMING	13	LUBRICANTS	86	WEAR	17
FRICTION	50	LUBRICATING	40	WEARING	8

LIST OF WORDS IN CLASS 40 - CERAMICS

WORD	COUNT	WORD	COUNT	WORD	COUNT
ALUMINA	12	ENAMEL	52	PASTE	16
BINDER	24	FIRED	23	PERLITE	17
BRICK	15	FIRING	48	PORCELAIN	11
BRICKS	28	GLAZE	25	POROSITY	16
CAO	33	GRAPHITE	17	REFRACTORY	53
CERAMIC	147	KAOLIN	14	SILICATE	17
CERAMICS	46	MGO	26	SIO	33
CLAY	32	MNO	17	SIO2	13
CORUNDUM	10				

APPENDIX E

A SAMPLING OF COMPOUND KEYWORDS CHOSEN BY CLASS

LIST OF CKW IN CLASS 1 - AERO

AERODYNAMIC WAVE
AERODYNAMIC LIFT
AIR FOIL
AIR FOILS
ANGLE OF ATTACK
BOUNDARY LAYER
LAMINAR FLOW
RATE OF CLIMB
SHOCK WAVE
SHOCK WAVES
TURBULENT FLOW
WIND TUNNEL

LIST OF CKW IN CLASS 19 - PCHEM

CLOSED SYSTEM
IDEAL SOLUTION
PHASE DIAGRAM
PHASE DIAGRAMS
PHYSICAL CHEMISTRY
RATE CONSTANT
RATE CONSTANTS
REACTION RATE
THERMAL STABILITY

LIST OF CKW IN CLASS 28 - PETROL

GAS PRODUCTION
GAS RESERVES
OIL FIELD
OIL PRODUCTION
OIL RESERVES
NATURAL GAS
NATURAL GASES

LIST OF CKW IN CLASS 82 - ORD

FIRE CONTROL
FLAME THROWER
KILL PROBABILITY
MINE LAYING
MINE SWEEPING
SHAPED CHARGE
SMALL ARMS
SMOKE SHELL

APPENDIX F

FINAL DEFINITION OF CIRC II CLASSES

CLASS	ABBREVIATION	DESCRIPTION	COSATI
1	AERO	Aerodynamics	1
2	AIRCRAFT	Aircraft Equipment and Systems	1
3	AG	Agriculture, Agronomy, Horticulture, Farming, Soil Science, Pests and Crop Diseases, Forestry	2
4	LIVESTOCK	Animal Husbandry, Stockbreeding, Livestock, Dairy and Milk Products, Domestic Animals and Pets, Game and Fish Management, Animal Diseases and Veterinary Medicine	2
5	ASTRO	Astronomy and Astrophysics	3
6	ATMOS	Atmospheric Sciences, Ionosphere, Meterology, Rain, Snow, Wind, Weather Forecasting	4
7	BIO	Biology, Botany, Zoology	6
8	BACT	Microbiology, Virology, Bacteriology	6
9	PHARM	Pharmacology and Toxicology	6
10	ILL	Human Illnesses, Diseases, and Ailments	6
11	MED/SCI	Medical Sciences	6
12	CLINIC	Clinical and Military Medicine, Paramedicine	6
13	PHYS	Physiology	6
14	MED-INST	Medical Equipment, Bioinstrumentation	6
15	PSYCH	Psychology, Parapsychology, Psychiatry	5,6
16	R & D	R & D Management and Resources	5
17	CYBER	Bionics, Cybernetics, Prostheses	6
18	CH-ENG	Chemical Engineering	7

CLASS	ABBREVIATION	DESCRIPTION	COSATI
19	PCHEM	Physical Chemistry	7
20	ANALY-CH	Analytical Chemistry and Quantitative Analysis	7
21	INORG-CH	Inorganic Chemistry	7
22	ORG-CH	Organic Chemistry	7
23	OCEAN	Oceanography	8
24	GEOG	Cartography and Geography, Geodesy, Topography, Surveying	8
25	GEOPHY	Geophysics, Geomagnetics, Terrestrial Magnetism, Geodynamics-Seismology, Earthquakes, Volcanos	8
26	GEOL	Applied Geology - Field Work, Geochemistry, Hydrology, Dams, Petrology, Limnology, Paleontology, Fossils, Glaciers, Snow, Ice, Permafrost, Stratigraphy	8
27	MINE	Mining Engineering, Economic Geology, Exploration, Ores, Minerals, Deposits, Mineral Dressing, Excavation, Boring, Drilling, Mine Working and Operations	8
28	PETROL	Petroleum, Oil and Gas Production and Distribution, Refineries; National and World Oil and Gas Reserves	8,21
29	EL-INSTR	Electrical Instruments - Electrical Networks and Circuits	9
30	EL-COMP	Electrical Components - Production of Electrical Accessories, Electrical Manufacturing Industry, Lighting and Illumination Transformers, Transmission Lines, Wires, Switches, Relays, Fuses	9
31	CPTR-HD	Computers - Hardware and Components	9
32	CPTR-PG	Computer Programming, Computer Software and Data Systems, Information Systems	9

CLASS	ABBREVIATION	DESCRIPTION	COSATI
33	ELECTRONICS	Electronics, Semiconductor Devices, Amplifiers, Wave-Forming Devices	9
34	EMAGTECH	Techniques of Electromagnetic Waves, Oscillations, Electromagnetic Radiation, Guided Propagation	9
35	POWER	Large Scale Power Generation, Distribution, and Control; Steam Turbines and Water Power	9,10
36	MOTORS	Motors and Electric Drive	9,10
37	BATTERY	Stored Energy and Power Sources, Batteries, Electrochemistry, Solar Energy, Thermoelectricity and Fuel Cells	9,10
38	FURNACES	Furnaces and Boilers, Electric Heating	10,13
39	OIL/LUB	Oils and Lubricants, Hydraulic Fluids	11
40	CERAMICS	Ceramics and Clay Industry, Refractories	11
41	GLASS	Glass Industry	11
42	CEMENT	Cement and Concrete	11
43	PAINTS/CTG	Dyes and Paints; Coatings, Colorants, and Finishes; Solvents and Cleaners	11
44	NF-MET	Metallurgy, Non-Ferrous Metals	11
45	F-MET	Ferrous Metals, Iron and Steel, Alloys	11
46	WOOD	Timber and Wood, Paper and Pulp	11
47	TEX/FIB	Textiles and Fibers; Clothing	11
48	RUB/PLAS	Rubbers and Plastics	11
49	MATH	Mathematical Sciences	12
50	CONSTR	Construction Industry; Construction Equipment and Materials	13
51	AIRC/HEAT	Air Conditioning, Heating and Ventilating; Heat Pumps	13

CLASS	ABBREVIATION	DESCRIPTION	COSATI
52	ENGINES	Internal Combustion and Other Engines	13
53	TRANS	Ground Transport and Transportation Engineering; Railway, Highways, Automobiles	13
54	CIV-ENG	Civil and Structural Engineering, Dams	13
55	PLANT-ENG	Plant Engineering; Containers and Packing, Warehouses, Depots; Assembly Lines and Production	13
56	FOOD	Food Technology, Food Industry, Beverages, Stimulants	13
57	FORGE	Forges and Forging; Tool and Die Making; Workshop Practice; Powder Metallurgy	13
58	MTL-HANDLE	Materials Handling; Hoisting, Cranes, Jacks; Mechanical Fixing and Attachment	13
59	ROLL/PIPES	Rolling, Drawing, Boiler-Making, Sheets, Tubes, Pipe Construction	13
60	MACH-TOOLS	Machine Tools, Planing, Milling, Grinding, Polishing, Shearing, Presses, Screw Cutting, Saws, Lathes, Drills, Punches	13
61	POWER-TRANS	Power Transmission, Bearings, Gears, Bushings, Cams, Clutches, Links, Linkages, Pulleys, Wheels, Chains	13
62	FLUIDS/PUMPS	Fluid Distribution, Storage, Containers, Pipes, Pumps, Filters, Tubing, Valves, Hydraulic and Pneumatic Equipment	13
63	NAV-ENG	Naval and Marine Engineering; Hydraulic Engineering, Ports, Harbors, and Coast Works	13
64	ENV-ENG	Environmental Engineering, Protection, and Pollution Control; Public Health and Safety Engineering	13
65	WELDS	Welding and Soldering	13
66	MTL-TEST	Material Testing and Physical Nature of Matter	11,13 14,20

CLASS	ABBREVIATION	DESCRIPTION	COSATI
67	LAB-TEST	Laboratories, Test Facilities and Equipment; Recording Devices and Instruments	14
68	G-MIL	General Military Activity, Training, Intelligence and Security	15
69	MIL-MAT	Military Material and Ground Equipment	15
70	MIL-OP	Military Operations, Defense, and Warfare (includes ASW)	15
71	CBR/NUC	CBR and Nuclear Warfare	15
72	MIS-TECH	Missile Technology	16
73	MIS/SYS	Missile Equipment and Systems	16
74	NAV/GUID	Navigation and Guidance, Direction Finding	17
75	DETECT	Magnetic, Acoustic, Infrared and Ultraviolet Detection	17
76	CTRMEAS	Electromagnetic and Acoustic Countermeasures	17
77	TELCOM	Telemetry, Telecommunication, Telegraph, Telephony	9,17
78	RADIO	Radio, Transmitters, Receivers, Television	9,17
79	NUC/MAT	Nuclear Fuels, Materials, Isotopes, Wastes, Byproducts	18
80	NUC-REACT	Nuclear Reactors for Large Scale Power Production and Propulsion	18
81	NUC-PHYS	Nuclear Physics	18,20
82	ORD	Weapons, Ordnance, and Ammunition	19
83	MECH	Mechanics, Measurement of Motion, Length, Acceleration	20
84	GAS/FL	Gas and Fluid Mechanics (Plasma Physics)	20

CLASS	ABBREVIATION	DESCRIPTION	COSATI
85	VIB/ACOUS	Vibration and Acoustics	20
86	OPTICS	Optics and Light; Photographic Techniques	20
87	THERMO	Heat and Thermodynamics	20
88	SOL-STATE	Solid State Physics	20
89	EMAG	Electricity and Magnetism	20
90	CRYSTAL	Crystallography; Diffraction	7,20
91	FUELS	Fuels	21
92	PROPEL	Propellants	21
93	SAT	Artificial Satellites	22
94	SPACE	Space Technology and Exploration; Rocket Technology	22
95	ECON	Economics and Finance	0,5
96	BUS	Business, Commerce, and Industry; Advertising and Marketing	0,5
97	COV/POL	Government and Politics; Propaganda	0,5
98	SOC-SCI	Social Sciences, Religion, Education, Humanities	0,5

MISSION
of
Rome Air Development Center

RADC plans and conducts research, exploratory and advanced development programs in command, control, and communications (C^3) activities, and in the C^3 areas of information sciences and intelligence. The principal technical mission areas are communications, electromagnetic guidance and control, surveillance of ground and aerospace objects, intelligence data collection and handling, information system technology, ionospheric propagation, solid state sciences, microwave physics and electronic reliability, maintainability and compatibility.

