INFORMATION RETRIEVAL
IN THE
BUSINESS ENVIRONMENT

Einar Nodtvedt

TR 80-447
December 1980

Department of Computer Science
Cornell University
Ithaca, New York

Information Retrieval in the Business Environment

Einar Nodtvedt

Visiting Fellow, Cornell University
(A.S. Computas, Hovik, Norway)

ABSTRACT

This paper outlines a possible approach to the use of advanced information storage and retrieval techniques for business correspondence. The present situation in the Office Automation field is surveyed, and the role of information retrieval in an integrated office information system is discussed. Business letter characteristics that might be useful for analysis and retrieval are described. The idea of a generalized business thesaurus containing standard phrases and locutions as well as synonyms and index phrases is presented. A technical solution to the analysis, storage and retrieval of business letters based on the concepts from the SMART-system is outlined. A description of the experiments performed so far follows. These include decomposition, analysis, storage, automatic classification and utilization of inter-letter references. The results from the experiments show that the use of automatic indexing and retrieval in the office is feasible and provides a viable alternative to existing manual business files. In conclusion, several future experiments are outlined.

Acknowledgements

This work has been made possible by grants from Det norske Veritas [of which A.S. Computas is the Data Division] and Norges Teknisk Naturvitenskaplige Forskningsråd [Royal Norwegian Council for Scientific and Industrial Research]. The author would like to express his thanks to the Computer Science Department at Cornell University for providing office space and computer resources during the academic year of 1979/80.

I am grateful to Dr. Cerard Salton and Dr. Harry Wu for guidance and assistance. I would also like to thank Dr. Thomas M. Lodahl, Professor of Administration, for his discussions of organizational design for office automation. Special thanks are due to Mr. Edward Fox for his help in finishing the manuscript and for his invaluable assistance during my year at Cornell.

Long Island, N.Y. - 30 July 1980

Einar Nodtvedt

TABLE OF CONTENTS

Information Retrieval in the Business Environment

Einar Nodtvedt

Visiting Fellow, Cornell University

(A.S. Computas, Hovik, Norway)

Summary of Report

This paper outlines the use of advanced information storage and retrieval techniques for business correspondence. This aspect of office information systems has so far been largely ignored both by researchers and vendors.

Chapter 1 gives the background for the interest in office automation and gives a short outline of the state-of-the-art both with respect to ongoing research and existing products in office automation. A description of existing, manual filing systems used in the office environment is given, and the expected benefits from automating the information retrieval process are described. The list of tools to be used includes the SMART-system, an advanced information storage and retrieval sytem developed for the library environment.

Chapter 2 describes characteristics of business letters that

might be of use during analysis and retrieval. These characteristics include: physical layout, letter type, style and standardized vocubularies. As an aid in the analysis phase the use of a business thesaurus is suggested. This thesaurus should contain words and phrases that are typical of business letters such as salutations, closes, courtesies etc. These will be used during an automatic analysis to extract useful information from the letters. Furthermore the thesaurus should contain synonyms and phrases relevant to the businesa that is using the system.

Chapter 3 outlines the main steps of a fully automatic analysis, indexing, storage and retrieval system. The analysis will start by using the business thesaurus to decompose the letter and extract the factual identifiers (i.e. author, recipient and date) from the "header" and "trailer" parts. The type of the letter may, in many cases, be determined automatically from the main text of the letter with the help of standard phrases from the thesaurus. The anslysis step will also search for phrases in the text indicating a reference to other documents (previous letters or enclosures), and phrases that do not carry any meaning during retrieval like courtesies and personal references. All these occurrences should be deleted before indexing starts. The inter-document references can be used to improve retrieval performance by transferring concepts from previous letters to the new letter after a reduction in "weight" (i.e. relative importance of terms). Standard automatic indexing, as used in the SMART-system, will be used to determine the concept terms describing the document content. These routines include the deletion of function words (i.e. "the", "a", "and", "but", "for" etc.) using a stop-list, and the reduction of words to stems by suffix removal. The indexing step will use the thesaurus to combine synonyms into classes, and to allow the use of phrases during indexing. The factual identifiers will be unweighted, while the content identifiers will be weighted using the IDF-method (Inverse Document Frequency); this assigns a high value to rare terms and a low value to terms occuring in many documents,

Chapter 4 describes the suggested storage organization. A combination of clustered and inverted files will be used. The inverted file will be used for the factual identifiers and the clustered files for the content identifiers. This will open the possibility of direct retrieval based on the factual keys, and will also allow range-searches (i.e. all letters in May 1975). The clustered files will give the opportunity of ranked output and a place-near logic for related documents.

Chapter 5 gives an overview of the retrieval part. This should preferably be interactive and accessible from the users normal work station. The suggested implementation has a user interface using natural language input. Output from the system should be given to the user terminal with options for hard-copy printing in case of CRT-terminals.

Chapter 6 describes the experiments performed so far on a collection of letters from a university environment. The experiments include the automatic analysis and decomposition of the letters. The output from this step is a list of factual and content terms describing the document. The collection has been analysed using several settings of various parameters that result in sentence and word-deletion, and the use of inter-document references has been investigated. Automatic classification of documents (clustering) has been done for different versions of the collection, and the results compared. The indication so far is that automatic indexing is feasible, and that the use of references improves performance.

Chapter 7 outlines future experiments that should be performed. These include a full scale implementation of a business thesaurus, and the utilization of letter type during retrieval.

Information Retrieval in the Business Environment

1. Introduction.

1.1. Office automation.

The last few years have seen an increasing interest in the productivity of the office worker. While blue-collar productivity has increased by more than 2% annually over the last two decades the white-collar productivity gain has been less than 0.5% (Diebold 79). This difference is also mirrored in the total investment in blue- and white-collar worker-places. In 1976 the average investment in a blue-collar work-place was much higher than in a white-collar work places ($24,000 vs. $2000, Purchase & Clover 76). The advent of the micro-processor is seen as a development that may alleviate this productivity problem. The micro-processor chip has made computing power cheap, and intelligent terminals and word-processors are beginning to enter the office environment. The emerging field of office automation is still in its infancy, but research is going on, and a steady rate of growth is predicted.

Office automation primarily tries to aid in the following four functions (Diebold 79):

- Text handling
- Communications
- Information storage and retrieval(IR)
- Data analysis

Research efforts in the field are directed both at technical as well as non-technical aspects of the new tools. In the technical area emphasis is placed on hardwsre (VLSI) and such computer science related topica as distributed systems, parallel processing, database security and integrity (Rothnie et.al. 80). The non-technical research concentrates on the psychological, sociological, organizational and economical impacts of OA (Lodahl 79, Rogers 80). Much research is also being done in the areas of office information systems modeling, office system design and implementation, and user interfaces (Bair 80, Ellis & Nutt 80, Palme 80).

While the researcher try to cover the whole field of office systems the vendors have been promoting the word-processor(WP) and other purely technical solutions (Vaskewitch 80). Whether packaged as a stand-alone or shared-logic system, WP-machines are sold mainly as typewriter substitutes to save typing time for the secretary. Most electronic mail systems are being marketed independently of text-handling and storage systems.

There have been few attempts to standardize equipment and interfaces, and the result has often been unsatisfactory as the new equipment is unable to communicate with the variety of different automatic systems slresdy found in most offices (i.e. typesetting, corporate databases, external databases, dats networks etc). A stand-alone WP-machine represents a fairly large investment, and it is ususlly not considered economically feasible to assign one machine to each secretary. The result has been typing-pools or equipment sharing, both methods which often result in poorer service to the professionals.

Most WP-systems employ very primitive methods for information storage and retrieval. This, undoubtedly, stems in part from the fact that the most common storage media for these systems so far has been the floppy disc with its limitations in size and flexibility. The most common access method to stored documents is through the use of document identification, either numeric or alphabetical.

Ongoing research and experience with the new systems indicate that it will not be possible to realize the full potential of the new technology without some integration between office systems and existing computer based database and communication systems (Morgan 76, Diebold 79). It is also becoming clear that not only the secretarial tasks may be automated, but that also clerical, professional and managerial workers may benefit from the new toola (Lodahl 79). This extension of the user group calls for improvments in communication and retrieval capabilities of the equipment. This will, hopefully, help bring the information retrieval aspect to the forefront.

1.2. Information retrieval in the office.

In every office the process of information retrieval is of the utmost importance. Information is a company resource, and the ability to utilize information efficently is directly related to the sucess of the firm. Office workers have a need for a multitude of different files at different times. In existing systems the individual worker will keep some information in a private file in his desk. The department or organization to which the worker belongs will keep information that is important to the business as a whole in central manual files or automatic databases. The individual worker might also from time to time require the use of external databases. These may be reference databases like the New York Times Information Bank, or numerical databases like Predicast etc. The proliferation of different information sources makes control of the information utilization extremely difficult and requires the user to get acquainted with a number of different retrieval systems both automatic and manual.

The organization of existing document filing systems will vary depending on their use. The private files in the desk of the office worker usually consist of a limited number of file-folders each marked with a subject heading. The larger department and organization files usually occupy a number of file cabinets and contains copies of all documents considered to be important to the organization. The documents are kept on files for varying lengths of time depending on their content and importance. The periodic purging of files to remove documents that no longer have any value is a time consuming job. It is, however, necessary to keep the files from growing too big. The files will typically contain forms, letters, memos, reports etc. in hard-copy format. The file cabinets used require large amount of space, and retrieval is often difficult. There exists a number of different filing strategies (Maedke et. al. 74) at varying levels of sophistication. However, no universal standard has been successfully enforced, and as a result filing practices vary widely from one firm to the other depending on the whim of the person responsible for filing and retrieving. Traditionally, large firms have employed a record manager to control the flow and use of information. The increased use of in-house database systems for formalized information like orders, personnel files etc. have led to problems in control by dispersing the information over a number of systems. The problems are exaggerated further by the use of different access and update methodologies.

To be useable for retrieval the documents must be assigned identifiers that somehow describe them in a meaningful manner. The limitation imposed by the need for physical storage of the documents (i.e. one document may only occupy one location), has resulted in simple filing schemes in which documents are classified under one main heading and are stored accordingly. This main heading may be the name of a person or an organization, or the main subject, as perceived by the indexer, of the document. Within the main heading the document may further be arranged according to date etc. To help retrieval some offices have implemented systems using card-catalogues to help retrieval on "secondary" indexes. This type of indexing is costly, and as there again are no universally recognized standards these systems tend to be closely dependent on the persons originally setting them up. They will therefore often be difficult to learn when new people take over the system, and the result is performance degradation.

1.3. Automatic information retrieval systems.

Morgan (76) cites three problems that must be overcome before fully automatic business retrieval systems becomes viable:

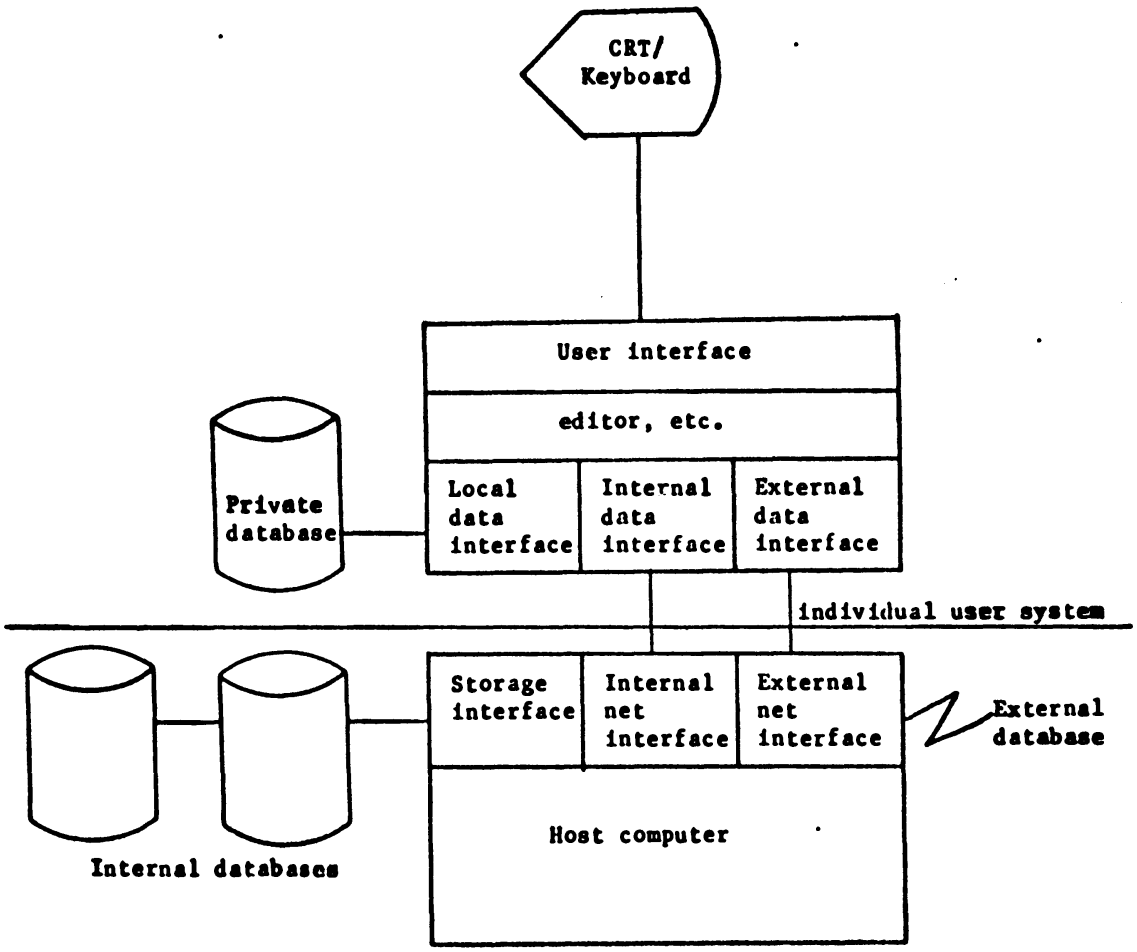
- Cost of storing full text
- Capture of documents
- Indexing

It has been claimed that the implementation of an office automation system may be cost-justified on the savings in filing space alone (Lodahl 79), and with the present trend of sinking hardware costs and growing costs elsewhere the first argument should no longer represent a problem. Document capture is still a problem, but with the introduction of in house word-processors and photo-typsetting most internal documents will soon be in machine-readable form. Electronic mail will take care of a large propotion of the incoming mail, and the remainder may be entered using OCR-equipment, voice recognition or retyping. When it comes to indexing this may be done by automatic means. Research in information retrieval of scientific papers shows that automatic indexing yields results that are comparable to that obtained from trained human indexers (Salton 72).

The storage of documents in machine readable form allows greater flexibility in storage and retrieval. The implementation of secondary indexes in the form of key-words is simple and this has been used for a long time in automatic retrieval systems in the libraries. When indexing it is useful to differentiate between content and factual identifiers. The content identifiers attempt to portray the meaning of the document. The factual identifiers, on the other hand, are objective properties of the document like: author, date written etc.

A fully integrated automatic retrieval system should preferably use only one terminal for all its functions (figure 1). This terminal should be placed on the office workers desk (or in his home), and be used for typing, editing, electronic mail and data analysis. Preferably only one set of commands should be used to access all information sources whether they be internal or external. This, of course, requires extensive programming of interface modules, but it would also save substantial amounts of time and money in user training. An integrated office terminal would allow the combined use of document and fact retrieval. This is a great help as it would allow the merging of address files with letters and the automatic production of collection letters, to mention but a few useful applications. A total system of this ambition is, however, beyond the scope of this work. The remaining part of this paper will concentrate on storage and retrieval of business correspondence.

Figure 1: User's work station.



1.4. Proposed line of work.

By combining the use of the new office technology with techniques known from the library environment and computer science it is possible to build a fully automatic office filing and retrieval system for business correspondence. The system will be based on the techniques that have been developed over the last two decades at Harvard and Cornell in the SMART-system (Salton 71). The SMART-system is a fully automatic information retrieval system that have been designed to work on collections of usually scientific papers. The system highlights are:

- automatic indexing
- vector-space model
- document clustering
- ranked output
- relevance feedback

By introducing these techniques it should be possible to reduce the amount of work done today in filing and retrieving, and to improve the efficiency in the retrieval process measured by the ability to find wanted documents.

The SMART-system was originally designed for the library environment where the collections usually are restricted to technical books and papers. A indexing program accepts as input abstracts of documents. The program breaks the abstract into single words, removes suffixes and deletes common function words. The words are sorted, assigned concept numbers through the use of a dictionary, and assigned weights based on number of occurrences in the abstract. The result of the indexing run is a vector of concept-weight pairs. This vector indicates the value of the different concepts, or terms, occurring in the text for retrieval purposes.

SMART uses a clustered file approach to storage of documents. This implies comparison of documents using a function that measure the amount of similarity between two document vectors. Documents having many concepts in common will get a high similarity value, and will be stored close together in clusters on secondary storage. To be able to access the stored documents a centroid is generated. This will contain all the concepts from the documents in the cluster as well as the document identification for each of the documents. In large files the centroids will again be clustered into supercentroids and hypercentroids therby generating a multilevel access tree.

In a business environment we will have a much wider range of input than in a library. The system will have to be able to handle information in different formats, i.e. letters, memos, meeting minutes, contracts, budgets, plans, status reports, forms etc. Of these formats the letter and memo formats are the ones differing most from the scientific abstract in form. Some of the other formats may be treated in the same manner as abstracts. But it is necessary to write a program that will pre-process the incoming letters and memos. This pre-processor will use the form of the letter together with the occurrence of standard locutions to extract the main factual identifiers. The parts of the letter containing the factual identifiers will then be removed before transferring the main body of the letter to the SMART-system for indexing. It will also be necessary to extend the storage and retrieval capabilities of the SMART system by incorporating the factual identifiers. The SMART-system does only allow retrieval of documents using the content identifiers. In a business environment the retrieval of documents based on factual identifiers will be of great importance.

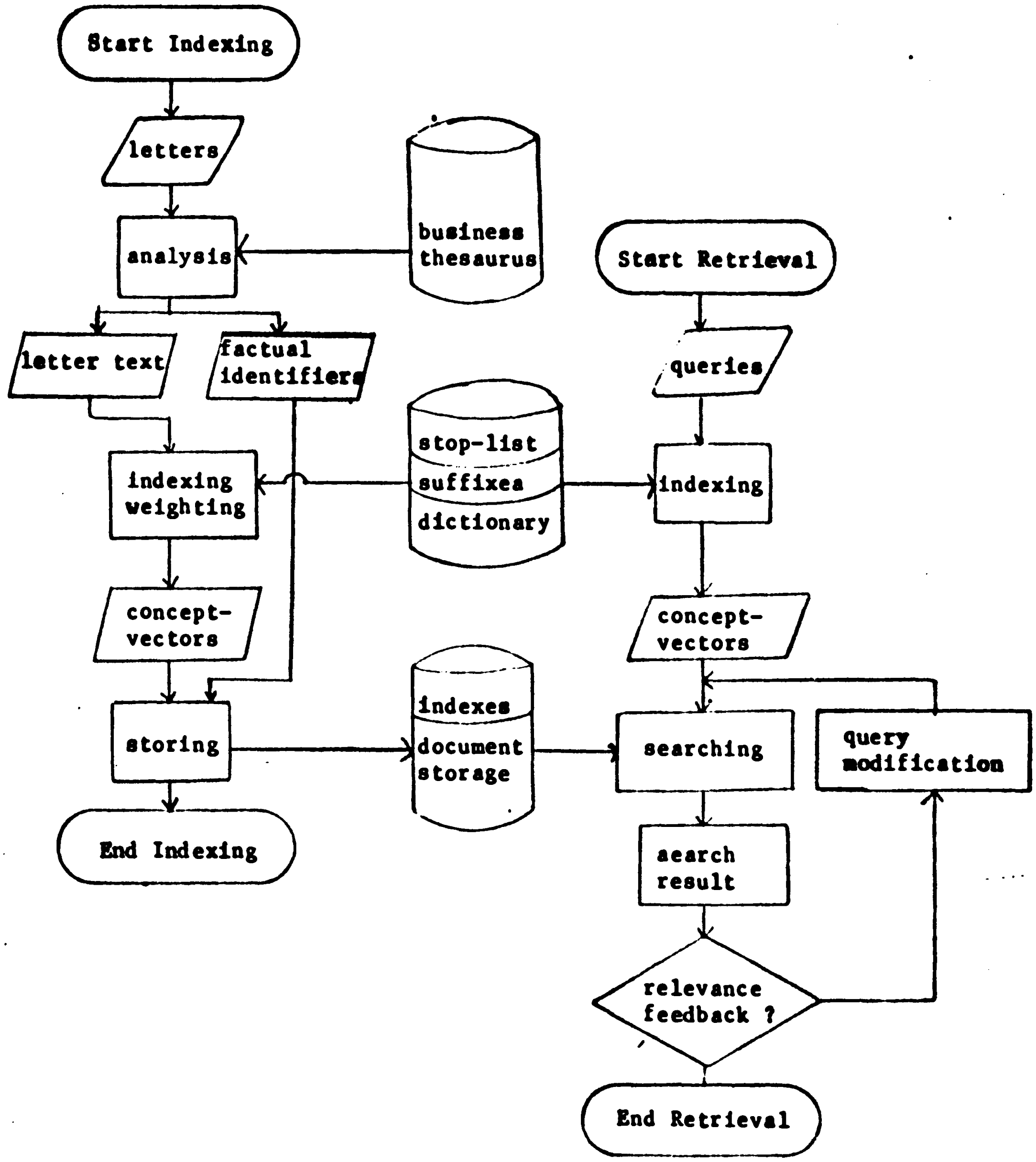
After analysing the document and assigning descriptors we will use the automatic clustering programs of SMART to classify the documents based on content. This means that documents that show great similarity will be placed in the same cluster and stored together on the secondary storage medium thereby, hopefully, reducing the number of data accesses needed to retrieve documents relevant to a user query.

The document retrieval process should preferably be executed interactivly from the user's work-station. The retrieval process will start with the user issuing a query in natural language form. This query is analysed by the indexing program, like the documents, and converted to concept vectors. This vector is then matched against the collection and the documents showing the highest similarity with the query will be displayed.

An automatic information retrieval system will allow the retention policy of the firm to be enforced with a minimum of effort. Monthly runs searching the files for letters that have passed their expiration date should be performed. The retention period will depend on type. (Maedke et. al. 74).

The SMART-system allows the use of relevance feedback. This implies that the user makes relevance judgments on the results presented to him from the initial search. These judgments can then be used to modify the query by up-grading the weight of terms found in the documents judged relevant, and by down-grading the weight of terms from the non-relevant documents. The refined query may then be resubmitted to the search program. The relevance judgments may also be used to modify the records in the collection by up- and down grading of good and bad terms respectively. In the initial system this feature of SMART will not be used, but in later versions this will form an interesting topic for further research. The complete system will look like the flow-chart in figure 2.

Figure 2: Proposed system flowchart.



1.5. Report outline.

This report describes the implementation of an automated office retrieval system using as a basis the SMART techniques and routines. Chapter 2 outlines some of the characteristics that we feel are important in the analysis of business letters. Chapter 3 gives details on a suggested strategy for this analysis. Chapter 4 describes the techniques used in automatic indexing while chapter 5 shows the retrieval process. Chapter 6 summarizes and discusses the results of the experiments performed to date and chapter 7 sketches the future experiments that should be performed.

2. Business letter characteristics.

2.1. Office environments.

Investigations done in the field of office automation (Lodahl 79) suggests that there are two extremes of office environments with markedly different characteristics. These two extremes are:

Custom	Production
Complex material	Simple material
Novel(new)	Routine(old)
Many languages	Few languages
Many styles	Few styles
No use of forms	Extensive use of forms
Ill-defined structure	Well-defined structure

The production office environment is normally characterized by well-structured and routinely executed tasks using a large number of pre-formatted and highly stylized letter types (orders, billing etc.). The information found in production type letters will in many cases already be part of existing database systems in the business (order-entry, accounts receivable etc). The need for retrieval of this type of information can usually be covered by existing computer systems. Examples of production type offices are insurance offices, sales offices, production plant offices etc.

The custom office, on the other hand, will deal with less well-defined types of letters (general requests, high-level decisions, new problems). These letters will cover a wide variety of language domains (law, computers, planning, public-relations etc) and styles. They will most often deal with information that is not found in existing database system and that is not likely to be entered into databases due to its inherent complexity. The only way to retrieve this information will therefore be through the correspondence files. This places great demand on the retrieval system as these letters may be of large value to the business. We are thus faced with the problem that the most important letters are the ones that pose the greatest analysis problems.

Normally a business office will lie somewhere in between the two extremes and will thus contain elements from both environments. We therfore suggest to design a system that will handle letters from both extremes.

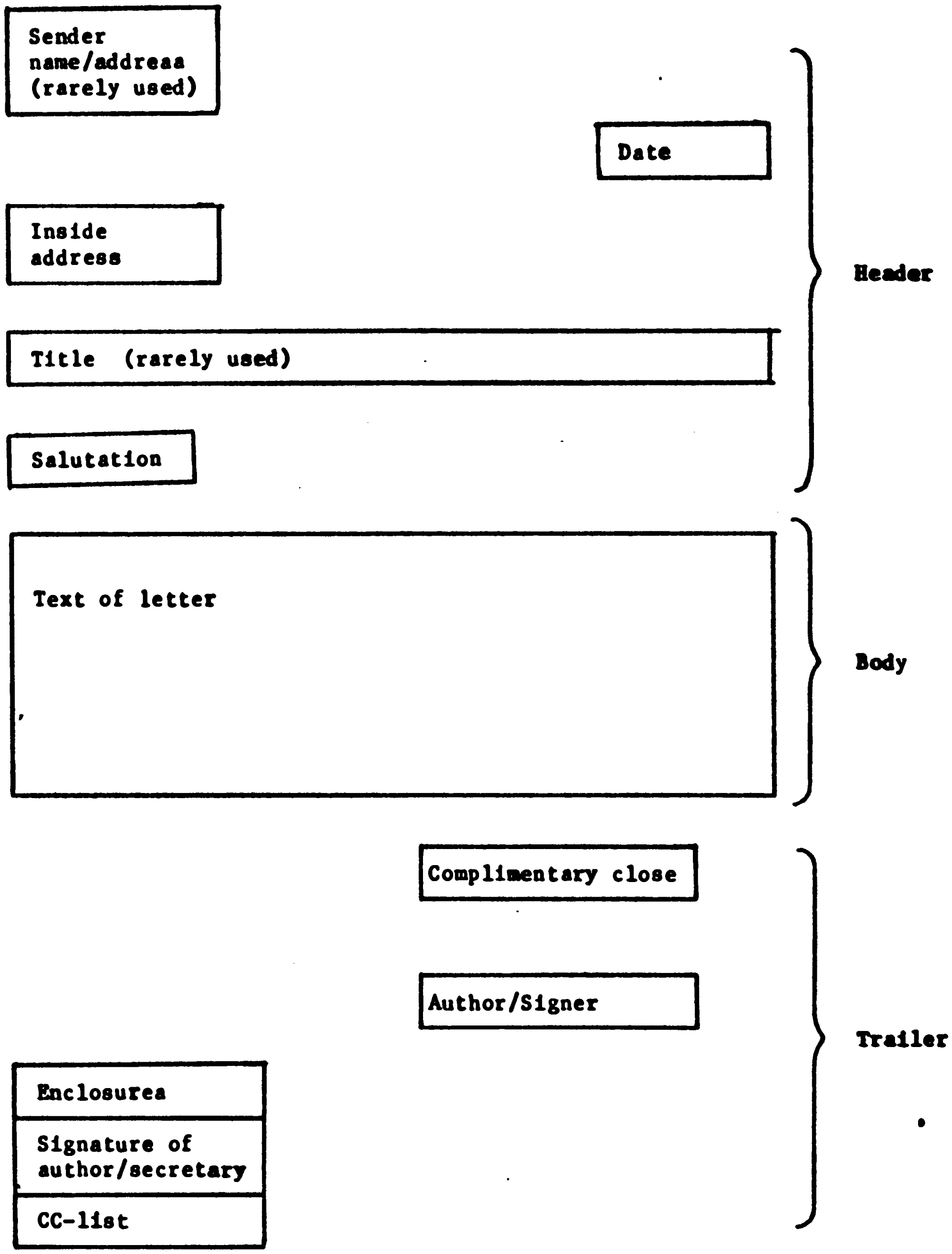
2.2. Letter characteristics.

Business letters are characterized by a number of different features that will help us in the analysis (Cresci 76). The normal target for automatic indexing has been the scientific abstract. These abstracts follow a fairly standardized format that include title, author and text. Abstracts are normally written in a strictly objective and detached style. A business letter, on the other hand, will have a more complex format including a number of extra factual identifiers. It will also be coloured by the relationship between the author and the receiver, the type of letter etc. These additional characteristics of the business letter compared to the abstract should preferably be utilized to improve the system performance.

2.2.1. Physical layout.

Business letters most often follow a rigid set of format rules. The rules that have significance in the analysis are shown in figure 3. The breakdown of the letter in three different main parts, namely:

Figure 3: Letter layout.



header, body and trailer allows us to treat the different parts using separate rules. The header and trailer will be used to find the factual identifiers, while the body will be analysed to find the content terms. There will, however, be some letters that do not conform to the basic rules, and means to handle these letters must also be given.

Electronic mail services will have their own format rules, varying from one system to the next. These rules will specify special fields for date, sender and recipient. The rules will be absolute within one system, and this makes the decomposition simpler than for normal letters. Memoranda will also follow a standard format that does not differ too much from the letter form, but this format will not be discussed in detail here.

2.2.2. Structure of body.

The content of the body of most letters follow a reasonably well-defined structure. Most letters start with some opening remarks. This may be a general presentation of the writer and his topic, or a reference back to an earlier letter (or conversation). Following the opening remarks is generally the main topic of the letter, and possibly also a number of secondary topics. Most letters end with some courtesies or phrases of a personal nature.

Even though these rules are generally applicable there may still occur letters or notes that lack all of the mentioned characteristics. In these cases the user should be given the opportunity to decide the fate of the letter: either discard it or index it manually. It is assumed that most often these letters will never be requested during retrieval, and that they may be discarded.

It should also be noted that while only the main concepts of the letter will be useful for content retrieval the other parts of the body may also be helpful during analysis. The reference part may give indication of theme by pointing to documents already analysed. The use of courtesies is an indication of the style of the letter and the degree of formality.

2.2.1. Letter types.

An important characteristic of a letter is its type. Determination of letter type will make it easier to pick out the main content identifiers. Finding the type also makes it possible to determine what part of the letter is the most important, and to increase the weights of the concepts found in that part. The type of the letter will also be an important search concept (i.e. All orders for widgets in January). The main use of letter type will be in the production environment where we have a set of well-defined and easily found types. In the custom environment it is more difficult to assign type to the letters as the letters will often fall into vague and unspecific classes that have little value during retrieval. In the custom environment, however, the normal method of automatic indexing will generally give good results.

While the classes in the production office will be applicable to all offices, the custom types will most probbly need to be specially designed for each separate office. We suggest to use the following main groupings of letter type.

- General
  - Narrative
  - Descriptive
- Requests/inquiries
- Replies
  - Favourable
  - Delay
  - Compromise/Alternative
  - Refusal
- Comments/Opinions
  - Positive
  - Neutral
  - Negative
- Reaction to received correspondence
  - Positive
  - Neutral (acknowledge)
  - Negative
- References/recommendations
- Rechannel to proper addressee

- **Purchasing (product/service)**
    - Request for information/offer
    - Sales letter/offer
    - Order
    - Answer to order
        - Inability to deliver
        - Delayed delivery
        - Suggest substitute
    - Shipment
    - Billing/invoice
    - Payment
    - Request status of order
- **Complaints/adjustments**
    - Request adjustment
    - Cive full adjustment
    - Compromise(partial adjustment)
    - Refusal
- **Credit letters**
    - Credit offer
    - Application
    - Give credit
    - Refusal

- Collection
    - Request payment
    - Payment
    - Request extension
    - Cive extension
    - Refuse extension
- Jobs (Type of job)
    - Application
    - Appointment for interview
    - Answer to application
        - Job offer
        - No job offer
    - Answer to job offer
        - Accept
        - Decline

It should be pointed out that one letter may fall into more than one of the above categories. The purchase and job classes should be further identified by the product or job referred to.

2.2.4. Style.

A complicating factor during the analysis is that the letters will vary in form or style depending on the relationship between the author and the receiver. This includes variance in formality and politeness. Furthermore the author's personal style will vary. Some authors will be humoristic, emotional etc. This variance in style calls for a synonym dictionary to map different forms with the same meaning into one standard form. In very formal letters we will find a number of courtesies that have little bearing on the content of the letter. These courtesies should be deleted, if this does not influence the retrieval performance negatively, to save space and resources during storage and retrieval. The style is not expected to play a great role during retrieval, and it should not be necessary to store a style indicator with the letter.

2.2.3. Business Thesaurus.

The automatic analysis of letters to determine important characteristics must depend on some sort of scanning procedure that reads the letters character by character and tries to match the text against significant words and phrases. In automatic indexing systems a list of index phrases and synonyms is normally sufficient. To analyse letters we also need to find the phrases and locutions mentioned above to determine letter type, to decompose the letters etc.

The need to search the letters to find special occurrences of text calls for some sort of dictionary or thesaurus containing all words and phrses of value. Such a business thesaurus will contain general words like salutations and closes occurring in all letters, and synonyms and index phrases specific to a given business. The business thesaurus should be able to match phrases even if they are discontinous. A list of the main classes of words in the thesaurus is given below:

Business specific:

- phrases	Allows the introduction of concepts consisting of more than one word.
- functional classes	Groups together functionally related concepts into one concept class.
- abbreviations	Mapping of the full text of commonly abreviated names or concepts into acronyms.

General:

- references	Phrases pointing to related documents.
- salutation & close	Phrases or words that define the transition from one part of the document to another.
- courtesies	Phrases that have little value during retrieval.

Combination:

- synonyms	Maps two or more words into one concept.
- type	Phrases that strongly indicate the main purpose of the letter.

Upon succesful match between the document word and the thesaurus entry a code found in the thesaurus will determine further actions. This may include deletion, substitution or addition of new concepts, as well as transfer of control from one main analysis segment to another. The design of a good thesaurus will not be a trivial task, but the payoffs should be substantial. For a preliminary list of entries in a business thesaurus see Appendix A.

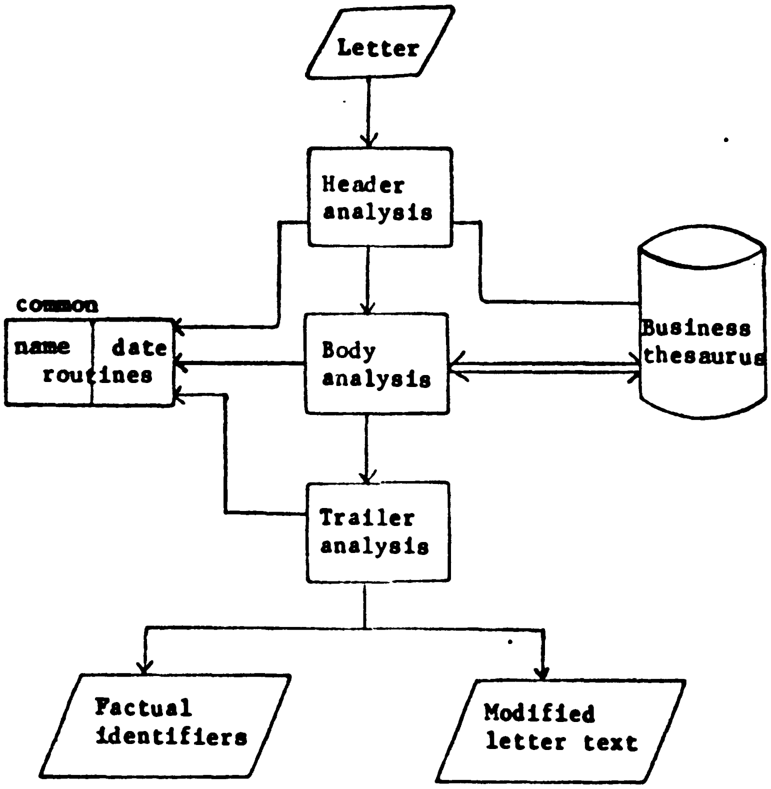
3. Analysis and indexing.

3.1. Introduction.

The analysis and indexing programs are used to determine the identifiers that describe the documents entering the system. The identifiers fall into two classes, factual and content. As there already exist programs for content indexing for the SMART-system, these programs will be used as they stand. The analysis specific to business documents will be handled by a special analysis program. This program will decompose the incoming letters, search the header and trailer for factual identifiers, and analyse the letter body for special occurrences useful in a business environment. The analysis program uses the business thesaurus described in chapter 2. and gives as output two files. One file contains the factual identifiers, and this file is transferred directly to the storage step. The other file contains a modified version of the body and is transferred to the indexing program for further decomposition (see figure 4).

A prototype of the analysis program has been developed on a PDP/11 computer using the programming language C. The program does the header and trailer analysis, but the full implementation of the body analysis routines using the business thesaurus has yet to be done; meanwhile, a simple approach to the body analysis is implemented that will search for and delete courtesies and other non-content bearing phrases. The present version of the program will not handle all possible input formats correctly, but it is felt that the results gained from this program will be a useful indication of the applicability of a fully developed program system.

Figure 4: Analysis program overview.



3.2. Analysis.

3.2.1. Decomposition.

The analysis program accepts as input the complete text of the letter. The program starts in header mode and stays in this mode until a salutation is found indicating the beginning of the body of the letter. The salutations are stored in the business thesaurus as are the complimentsry closes. The closes are used to determine the transition from body mode to trailer mode. The header and trailer are searched for factual identifiers and the values found are stored in special locations. The program searches for the end-of-letter character. When this character is found the program terminates the manipulation of the present letter. If one or more of the required factual identifiers are missing the user will be notified, and will be sble to intervene to insert the missing data.

3.2.2. Header and Trailer Analysis.

The analysis program searches the header and trailer to find the three main factual identifiers: author, recipient and date. The date is recognized using a table of month names, or abbreviations of month names, together with rules for placement of day and year. It is assumed that the date will occur in the header. The following formats are most commonly found in letters:

August 12, 1978	August 12, 78	August 12
12th August, 1978	12th August, 78	12th August
8/12/78	12/8 - 78 (European)	

All the date formats might be preceded by a place-name. Presently the program only recognizes the first two formats (American).

The recipient or addressee is found in the inside-address. We recognize the recipient by looking for a title using a table of possible occurrences. The title may in some cases be abbreviated (Professor - Prof.). In a more developed program the organizational affiliation of the recipient should be found, and possibly also the address. In some cases the name and address of the sender may be found in the header. If a search through the header discovers two name-address blocks we assume the first to be the sender. The header may also contain an indication of the main subject of the letter (Title: ..., Re: ..). If such an occurrence is found it will be transferred to the body-analysis program.

The trailer normally contains the author/signer of the letter. The author is assumed to be the first field following the salutation. So far only the name of the author will be used, but in a more developed system a look-up might be performed to determine the organizational affiliation of the author if this information is available from some data-base. Secretary, enclosure references and cc-list may also be found in the trailer.

1.2.1. References.

An important type of occurrence in the business letter is that of references to other documents, such as earlier letters or enclosures. This type of link is important during indexing for the following reasons:

1. Most letters are part of a correspondence chain going back and forth between two parties a number of times. During retrieval the user will often be interested in retrieving all the letters in one of these chains, and during analysis and storage this should be taken into consideration.

2. In a chain of letters we assume that there is a strong connection in content between letters adjacent in the chain. In many cases an answering letter or a letter accompanying enclosures will not itself give a clear idea of the main content of the letter. By going back to the referenced document it should be possible to determine letter subjects in these cases.

The backward references may be found during analysis by searching for the occurrence of a limited number of standard phrases (i.e. letter of ..., letter dated ..., note of .. etc) from the business thesaurus. The date may be expected to immediately follow such a reference, and can be determined using the same routine as was used during the header analysis to find the date of the letter. The name of the recipient and the date of the referenced letter is then used to perform a look-up in the document database to determine if the previous letter already exists in the files. The routine should be able to solve indirect references by the use of today's date, i.e. "your recent letter", "letter of last week" etc. should be used to perform an approximate date search. This option is not presently implemented.

After retrieval of a referenced document the information found may be used as follows. The document identification of the referenced document is stored with the new document and vice versa. This gives a double pointer chain allowing the movement both forwards and backwards in the chain after one document in the chain have been retrieved. In addition the concept vectors of the old document may be used to augment the new document vector by adding the concepts and weights. This method helps make documents in the same chain more similar, therby increasing the chance of entering them into the same clusters. It will also help determine the subject of documents only referring back to earlier documents and not explicitly stating the main subject. There is however a danger of swamping the letters in long chain by a steady build up in the number of concepts. To avoid this the concepts added into the vector will be downgraded to half their original weight. To limit the number of concepts the program deletes concepts with weight equal to or less than 1/4 of the minimum weight assigned by the indexing program (presently 12, see chapter 3.4). Thus terms occurring once in a document will not be carried forward more than one document unless the term reappears in the following letter (see figure 5). Experiments with other values for weight reduction and minimum weight should be performed.

Figure 5: Vector expansion for related documents.

	t1	t2	t3	t4	t5	t6	t7	t8	t9	t10
Initial document	12	0	24	0	0	24	12	0	36	0
Related document 1	0	0	12	12	0	36	24	0	12	12
Modified document 1	6	0	24	12	0	48	30	0	30	12
Related document 2	0	0	12	0	0	12	36	0	12	12
Modified document 2	0	0	24	6	0	36	54	0	30	18

1.2.4. Determination of type.

The basic purpose of a letter is to transfer information between a sender and receiver. The purely informative letters will be easily retrievable using the normal content identifiers, and for these letters the type is assumed to be of little importance. For the more specific and well-defined letter types from the production environment, like order, collection etc, the type will be of great importance during retrieval (i.e. Retrieve all complaints on a given product). The letter type lies in an area between factual and content identification, and experiments should be performed using the type in both these ways.

The type of the letter should be determined automatically by the analysis program.with the aid of the business thesaurus. The method suggested is based on the techniques described by Edmundson and Rush et.al. for use in automatic abstracting (Edmundson 67, Rush et.al. 71). The business thesaurus contains a set of cue-phrases (or words) that have been found empirically to be strong indicators of the type of letter in which the phrase occurs. To each phrase a weight is assigned signifying the effectivness of the phrase in uniquely determining the given type. The body analysis program will find all occurrences of type-phrases and rank them according to weight. The rank will be determined partly by the phrase itself and partly by use of location. This implies that cue-phrases found in the title of the letter will be considered most important. Cue-phrases found in the early paragraphs will be consiered to be more important than phrases occurring near the end of the letter, and cue-phrases found in the first sentences of a paragraph will be considered more important than phrases occurring later in the paragraph. After ranking, all types with weight above a certain predefined threshold will be assigned to the letter. The highest ranked type will be the main type, and any other types will be used as secondary types. This will cover the situation where a letter deals with more than one topic.

The number of different type indicators for a given type may be practically infinite, but most often the phrases are variations over a limited number of basic structures. Variations in choice of words, partial matching and discontinous phrases where non-significant words are found between the significant account for most of the variations. It is therefore suggested to use "templates" to define the structures for type definition. A templste is a pattern containing word-classes, indication of part weights for partial matching and indication of number of intervening words that are allowed between the significant words in the phrsse (Paice 80).

Alternative methods to the templstes include Knauss' heuristic weights or semantic networks (Knauss 80). The performance of the different methods should be compared to find the one that gives the best results.

3.2.5. Courtesies and personal references.

Formal business letters will usually contain a number of courtesies and personal references (depending on the relationship between sender and receiver) to round off the text. The occurrence of such phrases may be used to determine the degree of politeness and formality in the letter, but they are not considered to be of importance during retrieval. To reduce the size of the stored letter we therfore suggest deleting these phrases. The business thesaurus will contain a list of words and phrases that indicate the occurrence of courtesies. These words and phrases fall in two categories: The first category contains word that are strong indicators of non-significant sentences and results in the deletion of the complete sentence. The other category contains weaker indicators. This category will function as an additional stop-list (see chapter 3.3) and only the words and phrases themselves will be deleted.

3.2.6. Synonyms and abbreviations.

Experiments in automatic indexing show that the grouping of low frequency terms into synonym classea helps improve the retrieval performance of the system (Salton 75). In a business environment where the variations in style and vocabulary is even greater than for scientific abstracts, the use of appropriate synonyms becomes even more important. In addition to classes that map different words with overlapping meaning into one concept (unable - cannot, sorry - regret, paper - article etc.) the business thesaurus will contain special groups relevant to the business using the system. This may include classes like big customers, previous employees, trade conferences, journals etc. This special type of synonyms is incorporated into the dictionary under the header of functional classes.

A special type of synonyms is the mapping of full names into abbreviations and acronyms. In most businesses there will be found a certain number of special names and acronyms, and to improve retrieval performance it is suggested that all the full names are mapped into their abbreviations.

For all synonym occurrences the analysis program will add a synonym class identificator to the text of the letter. This identificator will later be assigned a unique concept number by the indexing program.

3.2.7. Index phrases.

Experimants in automatic indexing have also shown that the generation of index phrases by combining high frequency, co-occurring terms will improve system performance (Salton 75). The use of phrases make the content terms more specific and will help improve precision. In a business environment there will be a number of phrases of special importance, and these should all be entered into the business thesaurus. When the analysis program finds a index phrase it will add a phrase identification to the text of the letter. This phrases indicator will then be give a special concept number during the indexing phase.

3.2.8. Proper nouns.

Business letters will often contain proper nouns and these will be of special importance during retrieval. Product names, places and persons are among these occurrences. The analysis program will recognize those occurrences and help upgrade their significance by increasing their weight. The analysis program determines the occurrence of proper nouns by looking for capital letters ignoring the first capital letter in the first word of a sentence. This simple method is expected to be able to find practically all proper nouns in a document. To catch proper nouns occurring as the first word in a sentence the program could use the a list of especially important names stored in the business thesaurus. The classes and acronyms already described may also be utilized by the program.

3.3. Content indexing.

After having run the analysis step we transfer the modified body of the letter to the indexing program. The indexing program accepts as input a string of natural text (i.e. the body of the letter), and returns with a concept-weight vector. This vector consists of a collection of unique concept numbers identifying the terms occurring in the document, together with a weight indicating the number of occurrences of the term in the document. Presently the weight are given in multiples of 12. The program will split the string into single words and manipulate these words to assign concept numbers and weights.

3.3.1. Stop list.

The first step performed is the removal of all high-frequency function words. These are words that normally occur in most natural text documents and that therefore are of no help during classification and retrieval. The program utilizes a so-called stop-list of about 200 words. The stop-list includes:

Auxiliaries	can, will, may
Prepositions	in, of, at
Determinators	the, an, a
Connectors	and, but, either, or
Pronouns	I, you, me
Adverbials	since, when, what

It has been shown that a reduction of the total size of a text of between 30 and 50% is obtainable using this fairly simple scheme (van Rijsbergen 75).

3.1.2. Stemming.

In the text of a letter the same word may occur in a number of different forms. To map different forms of the same word into one concept we apply a stemming algorithm to remove the suffixes and reduce the words to common stems. The program uses a standard suffix dictionary containing the most commonly used suffixes in English. The algorithm starts by removing the longest possible suffix. If the remaining stem is too short (i.e. < 2 characters for some cases), the second longest suffix will be tried. After removing the suffix certain morphological analysis will be performed. This includes singling of double consonants at the end of stems, transformation of final i's to y etc. The algorithm used is due to Lovins(68). It is not possible to avoid a certain number of errors, but it has been shown that these are usually in less than five percent of the words processed.

3.3.3. Dictionary.

After removal of stop-words and stemming of the remaining words the word-stems are sorted and the number of occurrences of the different stems are counted. The stems are then matched against a dictionary. If the stem is already stored in the dictionary it is assigned the concept number corresponding to the stem. If the stem is a new one the stem is stored and a new concept number is assigned to the stem. Preferably the entries in the dictionary should contain numbers indicating the total frequency of the term in the collection. This will allow the calculation of IDF-weights (see 3.4).

3.4. Weighting.

The vector space model used in SMART requires that the concept terms all are assigned weights. These weights may be binary (1 - 0), signifying the presence or abcence of the term in the document, or it may be a integer weight giving a measure of the importance of the term in the document. A number of different weighting schemes have been used in the SMART system, most of which assign a value to each separate term and multiply this value by the number of occurrences of that term in a document to find the weight of the term for a specific document.

In the present project the so-called IDF value will be used. This assigns a high value to terms occurring in few documents, and a low value to frequently occuring terms using the following formula:

$$IDF_i = \log (N) - \log (D_i) + 1$$

where

N = total number of documents in collection

D_i = number of documents in which term i occurs

The weight is then calculated as

$$W_i = IDF_i \times TF_i$$ TF = term frequency in the given document of term i

The IDF weighting scheme is selected becauae it is easy to implement, requires limited resources, and yields results not far below the more advanced weighting schemes (Salton 75).

The business environment is expected to be very dynamic with a lot of new documents enetering the system and old documents being removed on a daily basis. The IDF values calculated above would require a full recomputation of weights for all documents having one or more terms in common with new documents entered or old documents deleted. This would clearly be impractical as this would not only require extensive computation, but might also lead to continous regrouping of document clusters. A dynamic version of the method is therefore suggested. This would require the document frequency of each term to be stored together with the relative document frequency at some initial point of time. The method also requires the system to know the collection size. Using the stored document frequency together with the collection size will give us a dynamic IDF-value, correct at the time of computation. Periodically a control run should be made checking the difference between the initial relative document frequency and the present, and if the difference for a certain minimum number of entries in the dictionary exceeds a given threshold a storage reorganization should take place.

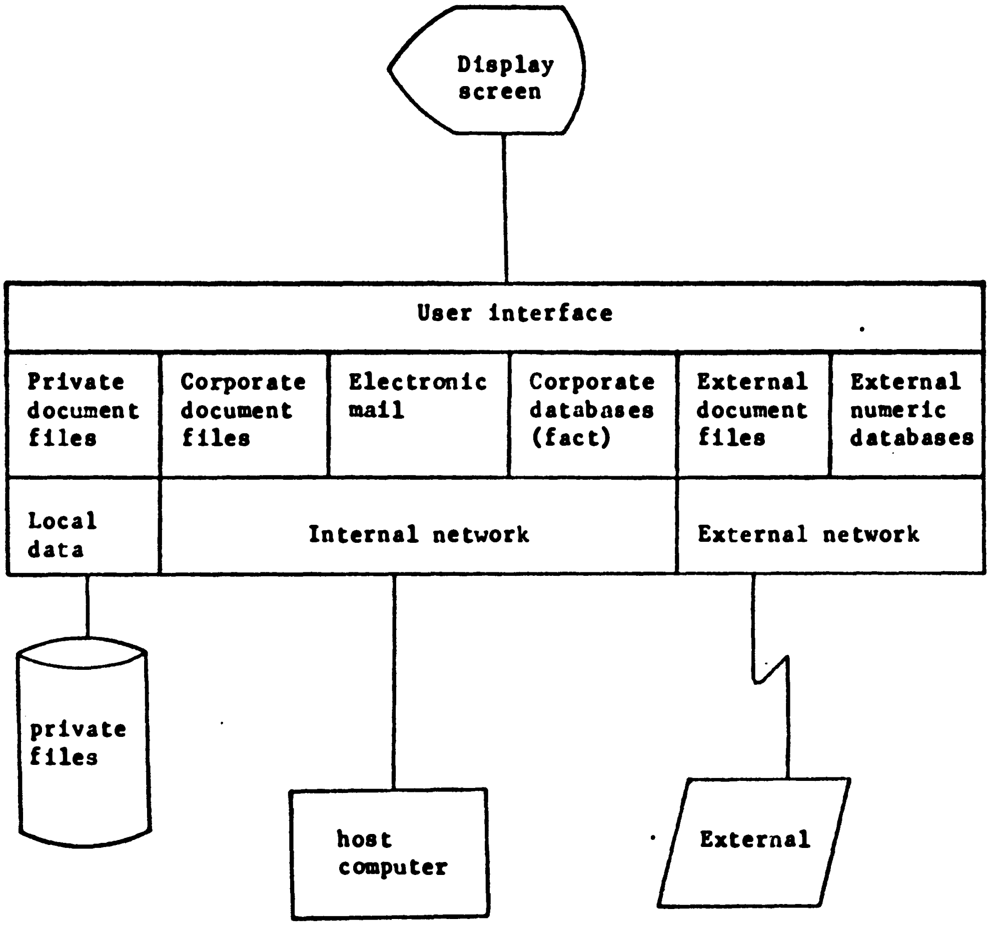
4. Storage organization.

4.1. Distributed systems.

In an operating office environment it is natural to disperse both the data and the computing power throughout the organization. The files will form a hierarchy from the personal files of the individual worker through departmental and corporate files. The placement of a given file in this hierarchy will determine the individual worker's access and control over the file. While the worker will have full control and access to his private files, the departmental and corporate files will only be partly acceasible depending on authority codes and access control. The personal files will be fairly small, but access will be fast and this is the natural place to put the documents directly relevant to the worker's daily job. In many cases it will be useful to duplicate documents, storing them both in the corporate files and in the private files of individuals to whom the documents are of special importance. Such a redundancy will save transfer time between sites during retrieval, but will pose new problems during updating. Extensive research is presently going on in this field (Rothnie et.al. 1980) and this aspect will not be further discussed here.

In a distributed system a communication network must exist to transfer documents and data among work-stations and to the outside world. An office information system therefore requires an extensive network; the work-stations must interface to a variety of network protocols and databases (figure 6). The problems involved in communications and the design of protocols and user interfaces are also targets of extensive research and will not be pursued further here.

Figure 6: Work station data stores.



4.2. Clustering.

A SMART-type clustering on the concept terms will be used during storage of the documents. This scheme implies the calculation of similarities between documents using some similarity function. The cosine function will be used. This function is defined as follows:

$$S(D_j, D_k) = \frac{\sum_{i=1}^{t} d_{ij} \times d_{ik}}{\left[\sum_{i=1}^{t} (d_{ij})^2 \times \sum_{i=1}^{t} (d_{ik})^2 \right]^{1/2}}$$

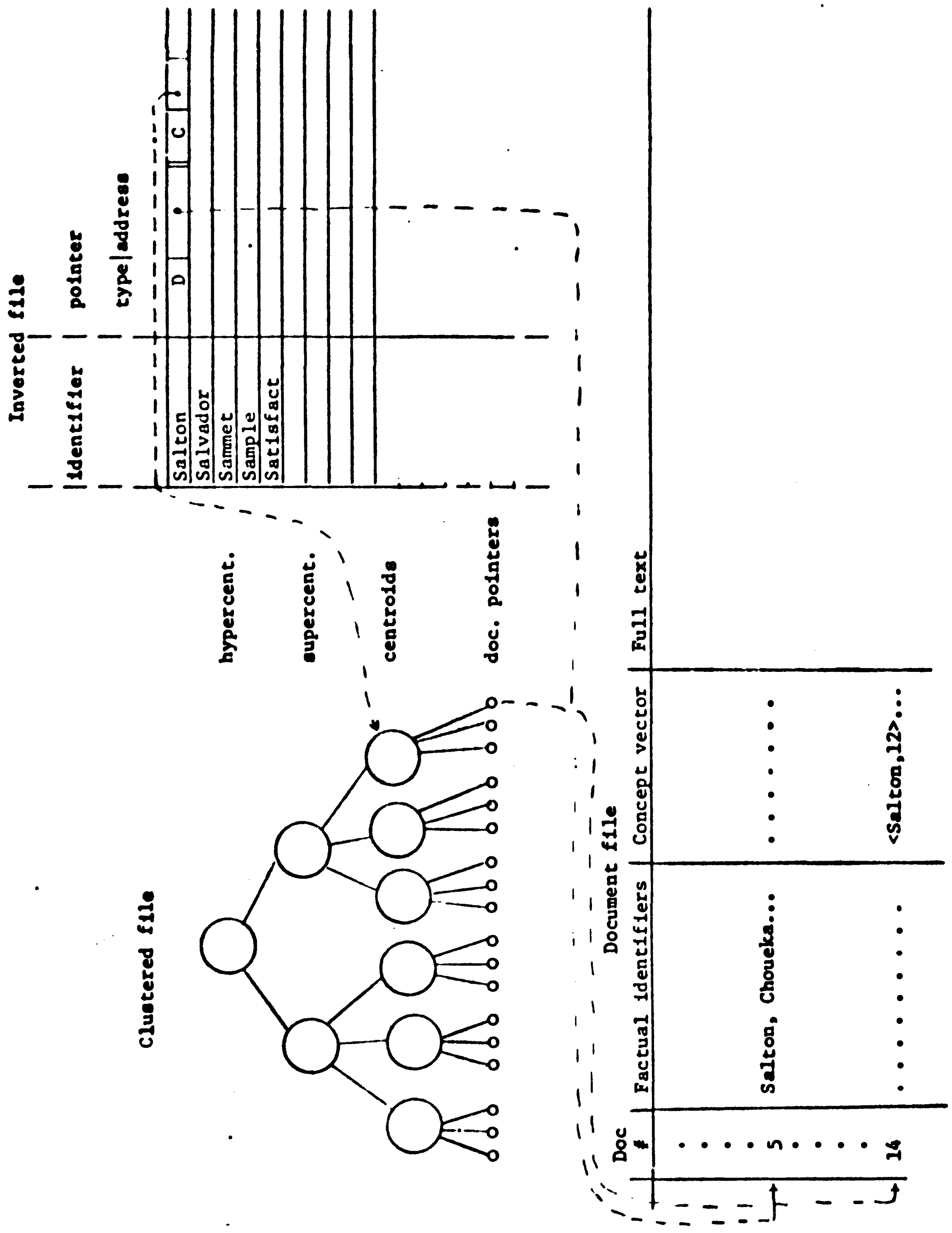
D_j = document no.

d_{ij} = weight of concept i in document j

t = number of different terms in system

Documents with similarities above a certain threshold will be placed in clusters. A physical cluster is a collection of document representations stored together on secondary storage. The placement together of documents in this manner reduces the number of storage accesses needed to retrieve all the documents in a cluster. To identify the clusters and gain access to the documents a centroid is generated for each cluster. The centroid represents the average of all the documents stored in the cluster. It contains all the concepts occurring in the documents belonging to the cluster with a weight for each concept equal to the average weight of the concept over the documents. The centroid furthermore contains a pointer to the physical storage location of each document stored in the cluster. In large files it will be necessary to group the centroids together in supercentroids, the supercentroids into hypercentroids and so on. This will give an access tree that may be used during searching (see figure 7). There exist a number of different clustering algorithms (Salton 71), and the user may specify parameters to control cluster size, similarity thresholds and amount of overlap

Figure 7: Suggested storage organization.



between clusters. A certain redundancy in the clusters has been found to be useful as some documents may fall in areas between well defined clusters. By introducing such documents into more than one cluster we may save searching time and accesses.

In addition to centroids corresponding to the physical storage of the documents we may also have secondary or virtual clusters. In these, clustering is performed using other clustering criteria than for the physical clusters. The documents belonging to virtual clusters will normally not be stored together as the physical storage is governed by the primary clustering criteria; hence the number of accesses needed to retrieve all documents in a cluster will rise. The secondary clusters may, however, be helpful in making possible searches otherwise impossible. The choice of criteria to be used for physical and virtual clustering should be made depending on the primary and secondary access paths relevant to the system.

In this project the primary clustering criterion will be the concept terms derived from the automatic indexing process. Secondary clustering should be attempted using letter type, letter chains and factual identifiers. The results obtained using the various cluster criteria should help in designing an optimal storage and retrieval system.

Figure 7: Suggested storage organization

4.3. Inverted files.

Retrieval based on the use of factual identifiers like author etc. differs somewhat from content retrieval. Whereas content retrieval is more vague and inexact, factual retrieval is more absolute. It is therefore natural to implement retrieval on factual identifiers with the use of inverted files. During a search for letters from John A. Jones in November 1979, letters from John. A. Smith in November 1978 will generally be non-relevant. Using a clustered file strategy this latter letter could well have been retrieved due to the fact that it has 3 out of 5 identifiers in common with the query. The factual identifiers should further be retrievable using range searches. That is searches where the user inputs a range of values that all represent a match (i.e. all letters in May and June 1978). This type of retrieval will not be practical using clustered files. A possible implementation scheme that will allow retrieval on the criteria outlined above is to use a sorted, index-sequential storage organization (see figure 7). The problem with this method is that it is inflexible and highly susceptible to skewed distributions of incoming identifiers. A possible solution is outlined in section 4.4.

4.4. Refinements.

To improve the performance of the systems the following methods will be considered.

- Low-level cluster inversion
- B-trees
- Soundex codes

To speed up search times by avoiding searching through deep tree structures for the clustered files, it is possible to invert on the low-level clusters. This method entails building an inverted file that does not point to the documents containing a given concept, but instead to the low-level centroids in which the concept is found. This method will limit the length of the search, while also limiting the size of the inverted file. The result will be a combined system incorporating the advantages from both clustered and inverted systems, while at the same time avoiding the disadvantages.

One of the main problems with sorted, index-sequential files is the inflexibility of the system. Insertion of new identifiers is time consuming, and if the new identifiers have a skewed distribution the index will very easily become unbalanced and require a complete reorganization. The use of B-trees will help alleviate this problem. In B-trees the routines used during insertion and deletion of concepts guarantees a balanced tree after the operation is finished. This will give constant search times in all parts of the tree, and complete reorganization will never be required (Bayer & McCreight 72). B-trees might also be used to store the cluster file tree.

Searches using the names of individuals and organizations will often be hampered by different methods of writing names that sound similar (Smith, Smythe etc.). As a result of this problem searches may miss documents that are relevant to a query. To avoid this a special routine that transforms all names into codes depending on their pronounciation has been devised. This SOUNDEX code will keep the first letter, remove most vowels and double consonants and map similar sounding letters into one occurrence. The use of SOUNDEX will be of great importance in a business environment where a large part of the retrieval will be based on names.

5. Retrieval.

5.1. Query formulation.

Fast access and ease of use are of the utmost importance in a document retrieval system. The aim is to replace existing manual systems; and to be able to offer a better service than what is possible today, on-line searching must be the standard mode of work. The user should be able to retrieve the documents from the same terminal that is used during generation and storage of the documents.

It is also of great importance that the retrieval process be user-friendly, and that it does not require extensive amounts of initial training. In the case of internal databases distributed over a number of different locations, the user should not have to know where in the system the documents he is looking for are stored. It should furthermore not be necessary to have to learn a number of different query formulation languages to be able to retrieve relevant information from separate systems. In the case of fact retrieval, self-instructing commands and operations should be provided for the casual user. Merging of data from fact databases and text from word-processing machines should be easily done.

This paper will concentrate on the retrieval of documents, and even though fact retrieval will be a central function of an intergrated office system, this aspect will not be discussed further. A natural language interface, allowing the user to formulate the queries in plain English is assumed to be the best solution. This requires minimal training, but demands a highly sophisticated user interface. A possible exception to the need for natural language queries is handling the factual identifiers. The need to specify range searches and to differentiate between factual and content identifiers may be easier to satisfy with the help of some predefined form. A sample set of queries presuming a natural language interface is shown in figure 8.

Content query:
    Application for funds to produce film on SMART.

Factual query: .
    Letters to/from Choueka in 1975

Combined query:
    Letters to Arden on COSERS-panels

Range search:
    Letters about SMART and automatic indexing with date
    between October 15, 1975 and March 1, 1976

Figure 8. Sample queries.

A function that is common in libraries, but that will be new to the office environment is the use of SDI-searches (Selective Dissemination of Information). This method implies that it is possible to define user search profiles containing topics and persons of special interest to the individual user. These profiles will be stored in the computer, and each new document will be matched against the profiles upon entering the database or batch comparisons will be made periodically comparing all newly entered documents with the profiles. Documents matching a profile may then automatically be transferred to the person owning that profile.

therby guaranteeing that everybody concerned with a certain product or project receives all relevant documents promptly. To avoid unauthorized personnel accessing classified documents a security code may be attached to each person, profile and document.

In the cases where the search is performed against external databases the user will have to know which database he is searching to gain access, but the query formulation should still be done using a natural language format. The translation to system dependent queries should be done automatically by an interface module in the work-station.

5.2. Searching.

The actual database search will depend on the type of query and the database to be examined. For searches against local databases where the queries only contain content identifiers the traditional SMART search routines will be used, with some modification to allow direct retrieval through an inverted file on the low-level clusters. Pure factual queries will likewise employ a traditional inverted file search strategy, somewhat modified by the presence of range-searches. In a combined query containing both types of terms we suggest to try some optimization. First the routine should perform a look-up in the inverted files. If the resulting number of documents are small enough all the documents will be retrieved and a similarity computstion will be performed on this subset of the collection. If the number of retrieved documents from the inverted files are great, however, we will perform a traditional content search and merge the resulting lists afterwards. The heuristic used in choosing when the number of documents are too great to allow similarity computation on the subset should be determined through experiments.

For external databases the search strategy will depend on the system used and the strategy can not be controlled from the terminal by the user. If possible, however, the database interfaces in the user terminal should try to make use of some limited query-optimization.

5.3. Presentation.

Using the SMART-philosophy allows us to present the result from a local search (containing content identifiers) as a ranked list of documents in descending order of similarity with the query. The initial display should contain a document identification and the main factual identifiers. Based on this list the user should be able to determine which documents are relevant to his needs in most cases. If this information is not enough the user should be able to select the individual documents for full-text display on the screen. After having found the relevant documents the user should be able to browse through them on his terminal, print out hard-copies on a nearby printer or use the text as a basis for producing new documents.

For queries without content identifiers the documents will be presented in order of retrieval. A possible extension would be to weight the factual query terms and present the documents ranked based on the weight of the matching factual terms. Given the nature of factual retrieval this method is not considered practical in this system.

The database system used will control the presentation when external databases are searched. The standard file organization in existing document databases is that of unweighted, inverted files. It is therefore not possible to rank the documents without the user assigning weights to the query terms. In this latter case, however, a ranking might be performed, but this would entail a fairly sophisticated system interface where the merging of the inverted lists for the individual concepts entered are performed by the user terminal instead of at the off-site computer. This type of ranking has been explored by the SIRE-project group.

5.4. Query and document modification.

The SMART system group has experimented heavily with query improving methods, one of the most important being relevance feedback. This method allows the user to react to the result of an initial query by marking documents that are relevant. The system then automatically reissues the query after modifying it in a way that hopefully will improve performance. This may include upgrading the significance of terms in the relevant documents, and a corresponding down-grading of the terms found in the non-relevant documents.

The user's relevance assesments might also be used to dynamically modify the collection by grouping documents that were judged relevant to the same query closer together and by moving non-relevant documents away from the relevant. This approach will be better considered in a fully implemented system where a history of queries and relevance assessments would be available.

6. Summary of experiments performed.

6.1. Test collection.

The test collection that have been used so far in the experiments consists of 134 documents from a university environment. The subject matter of the letters concentrates on SMART related topics including: requests for visits, exchange of ideas, publication of papers, grant proposals etc. The letters may best be classified as belonging to the custom-type environment, though some production-type letters also exist. The letters vary in length, structure and style. Some 75% of the letters are part of a chain of two or more letters exchanged between the SMART group and the outside world, while the remaining letters are stand-alone documents (figure 9). The relationship between documents is determined based on the occurrence of explicit backward references from one letter to a previous one.

The experimental runs used two different query sets of 19 and 24 queries respectively. The queries were formulated based on some knowledge of what was stored in the database. While this method would be considered inappropriate in a traditional reference environment, it is felt that this is the way queries actually are formulated in a business environment. The searcher will in many cases have seen the document he is looking for before, and he will use this knowledge in formulating his query. The queries vary in type from very specific, having only one relevant document, to broader, with up to ten relevant documents. The relevance assessments were made by comparing all queries against the complete document file. The queries have from four to ten query terms.

Figure 9: Test collection characteristics.

Inter-document relations in test collection.

Key: number means document identification

"-" means related

"," means unrelated

1-2,3,4,5-6-7,8-9,10,11,12,13-14,15-16-17-18,19,20-21-22,23-24,25-26,27,
28-29,30-31,32,33-34,35-36-37,38-39-40,41-42,43-44-45-46,47-48-49,50-51,
52,53-54,55-56,57-58,59-60,61,62-63-64-65,66,67-68-69,70,71,72-73-74-75,
76-77-78,79-80-81-82,83,84-85,86,87,88,89-90-91,92-93,94,95-96,97-98,99,
100,101-102-103,104-105-106-107,108-109,110-111,112,113-114,115,116,
117-118,119-120-121,122,123,124-125,126,127,128-129,130-131,132,133,134

Statistics.

Single documents: 30

Chains of length 2: 25

Chains of length 3: 10

Chains of length 4: 6

The experiments so far have only considered content-terms, and the queries do not contain factual identifiers. In related experiments using the same database the queries have been expanded to include factual identifiers as well. These experiments have not yet been completed; preliminary results indicate that the number of documents and queries is insufficient for accurate comparison of efficiency and effectiveness of differing storage and retrieval methodologies.

6.2. Analysis.

The analysis program is a pre-processor that scans the incoming letters searching for occurrences found in the business thesaurus. When a match is found, the program acts depending on the code associated with the thesaurus entry (i.e. synonym, abbreviation etc). The program is written in C under UNIX on a PDP-11/60. So far only a simple analysis is performed, splitting the documents into parts and determining the factual identifiers. A full scale implementation would require a completely developed business thesaurus and a more refined analysis program.

6.2.1. Decomposition.

The analysis program correctly decomposes all the letters using the salutations and complimentary closes found in the business thesaurus (see appendix A). The removal of the header and trailer from the letters reduces the size of the text that are sent to the indexing step by about 25%. A large part of this reduction stems from the removal of leading blanks from right-justified fields like date, signature etc. The rest comes from the deletion of addresses, organization names, salutations and complimentary closes. While the savings would be less substantial in a environment where the original text is compressed using some packing algorithm, there would still be significant savings.

6.2.2. Factual identifiers.

The analysis program correctly finds the following three factual identifiers: date, author and recipient. The date is found in the header by searching for occurrences of month names either written in full, or abbreviated (i.e. Aug.). So far only standard American format (i.e. June 27, 1980) is recognized, but other formats should not represent any problems. The receiver of the letter is found in the inside address using title as an indicator. The author is found in the trailer immediately following the complimentary close. In a operational system more information might be extracted, especially from the inside address and trailer. This includes organization of receiver, address of receiver, typist, cc-list etc. For the present experiments, however, the determination of the three most important identifiers is deemed sufficient. The program will give a message to the user when an end-of-letter is determined before all factual identifiers have been found. In an interactive environment this facility will give the user the opportunity to enter factual identifiers that the analysis program has been unable to find.

6.2.3. Deletion.

The business thesaurus contains phrases that are used during analysis to determine the decomposition of the letter, phrases that reference other letters, phrases of courtesy and personal references.

These phrases are useful during analysis, but are not expected to contribute significantly to retrieval performance. Experiments have been performed deleting these phrases and a 10% reduction in the size of the body of the letter has been found. This reduction is accompanied by a reduction of less than 5% of the dictionary. The deletion does, however, reduce the retrieval performance. Figure 10 shows the recall and precision graphs for the test queries with and without the "empty" phrases. The performance degradation arises from deletion of important words. With a better thesaurus that eliminate unecessary deletion this problem should be removed.

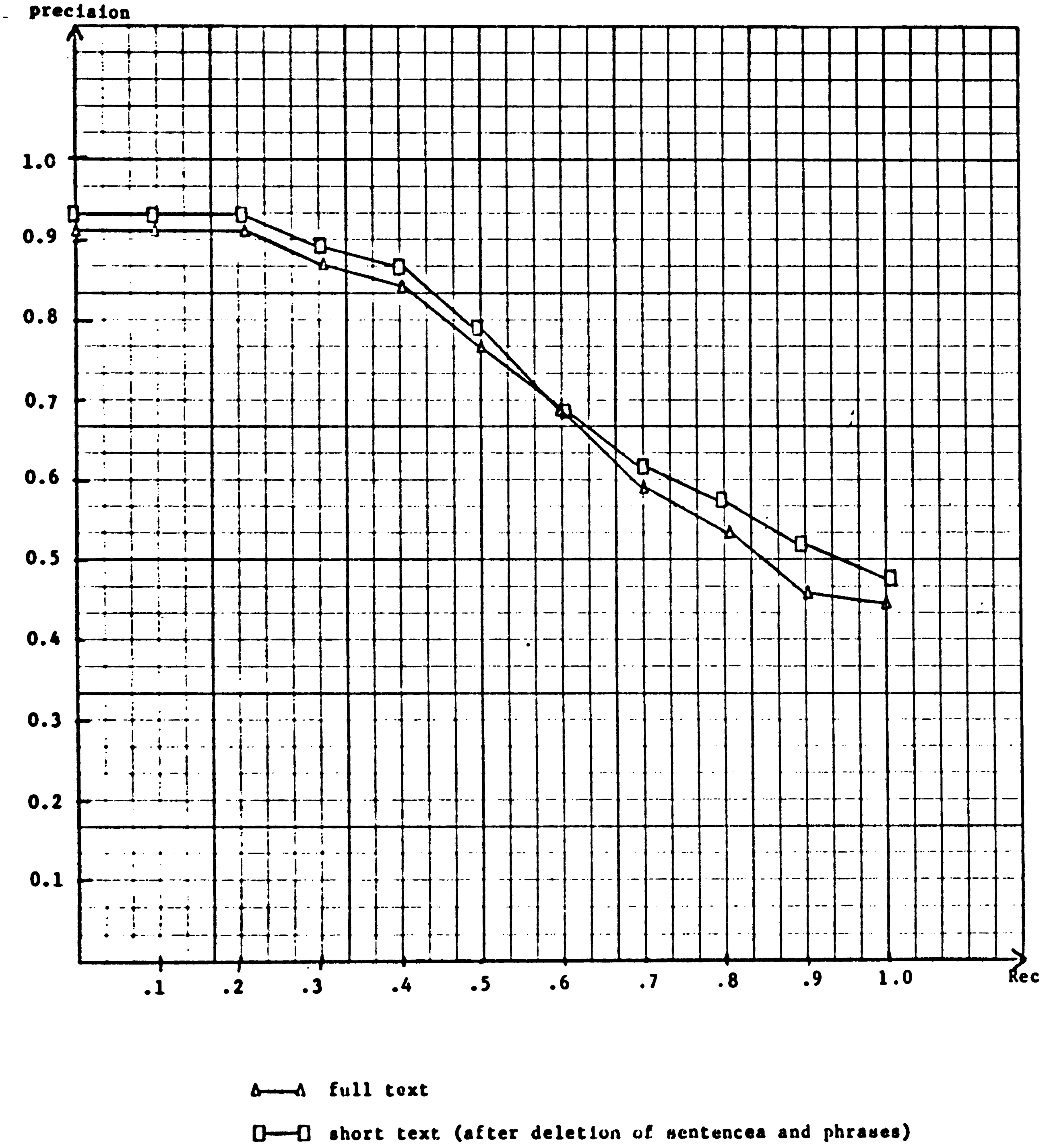
6.3. Indexing.

Content indexing has been done using existing SMART related programs. The programs do not consider phrases and synonyms, however, and we would therefore suggest implementing these features using the business thesaurus and the analysis program. The indexing program does have other shortcomings and should preferably be re-programmed. It only accepts upper-case characters, the stemming algorithm sometimea gives more than one stem for the same concept and the dictionary does not contain any frequency count for the individual entries.

6.4. Weighting.

The use of term weights has been shown to have significant impact on retrieval performance in traditional information retrieval systems using ranked output (Salton 75). Experiments have been performed that compare straight term frequency weighting (i.e. the weight of a term in

Figure 10: Effect of sentence deletion.



a document is only dependent on the number of occurrences of that term in the document) with IDF weighting where the weight is inversly proportional to the number of documents in which the term occurs. The results of the comparisons, for this collection, are shown in figure 11. As can be seen, IDF-weighting is superior to frequency weighting. This result has been shown to be statistically significant.

6.5. Relevance feedback.

Experiments have been performed using relevance feedback. Both positive and negative feedback were used. Positive feedback increases the weight of the query term found in documents that are considered relevant, while negative feedback decreases the weight of terms found in the non-relevant set. The results are outlined in figure 12, and show a statistically significant improvement.

6.6. References.

Backward references from one letter to its predecessor have been used in the following manner. All content terms from the original letter have been transferred to the second letter in the chain after reducing the weight to a half of the original. Weights of 1/4 of the original minimum weight (i.e. 12) will be deleted. In chains of more than two letters this will include terms from one or all of the predecessors depending on whether the earlier terms get used again. The resulting document vectors are somewhat longer than those in the system without references. The performance with the references has been compared to that of the original letters and we have found a certain

Figure 11: IDF weighting.

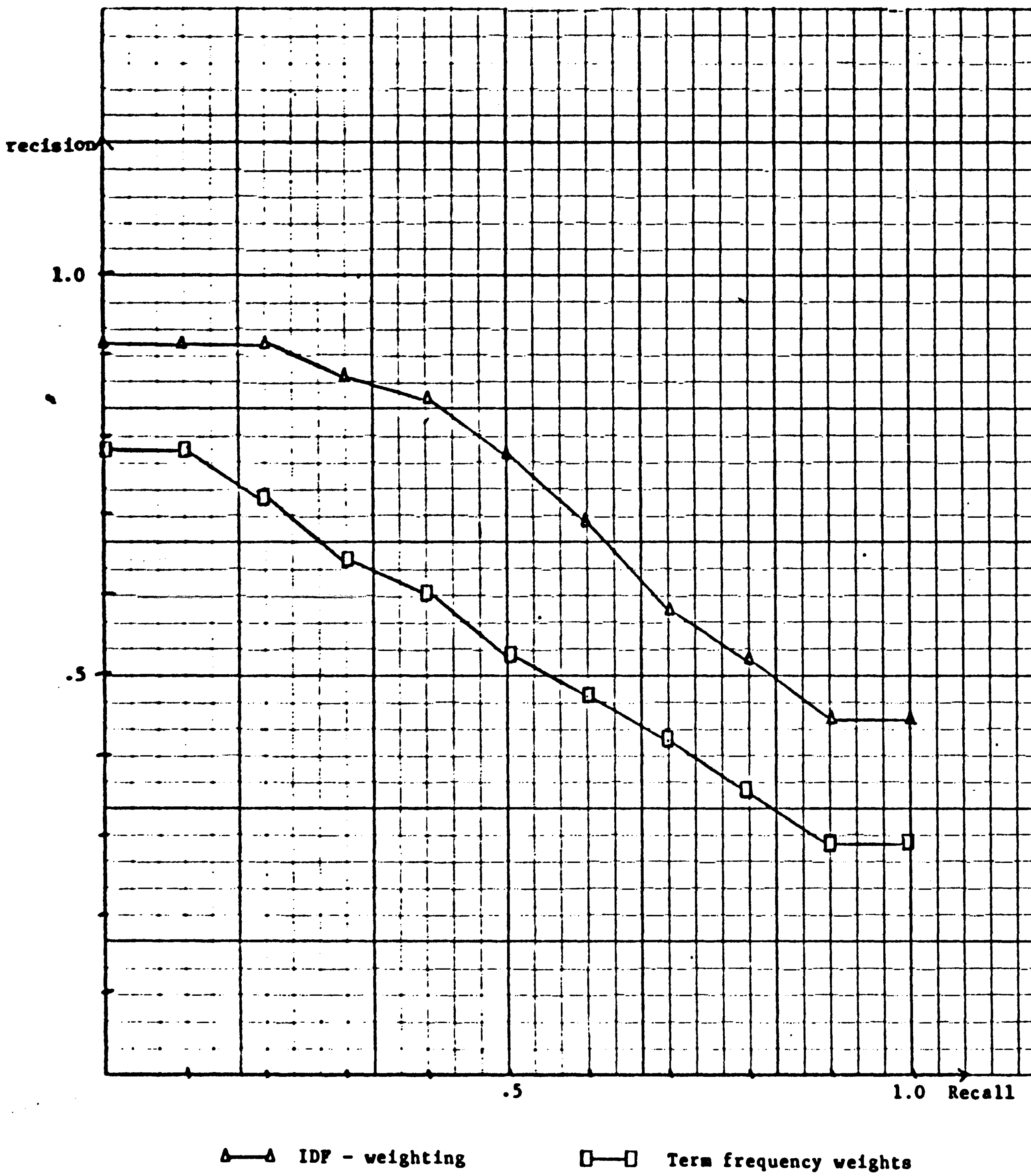
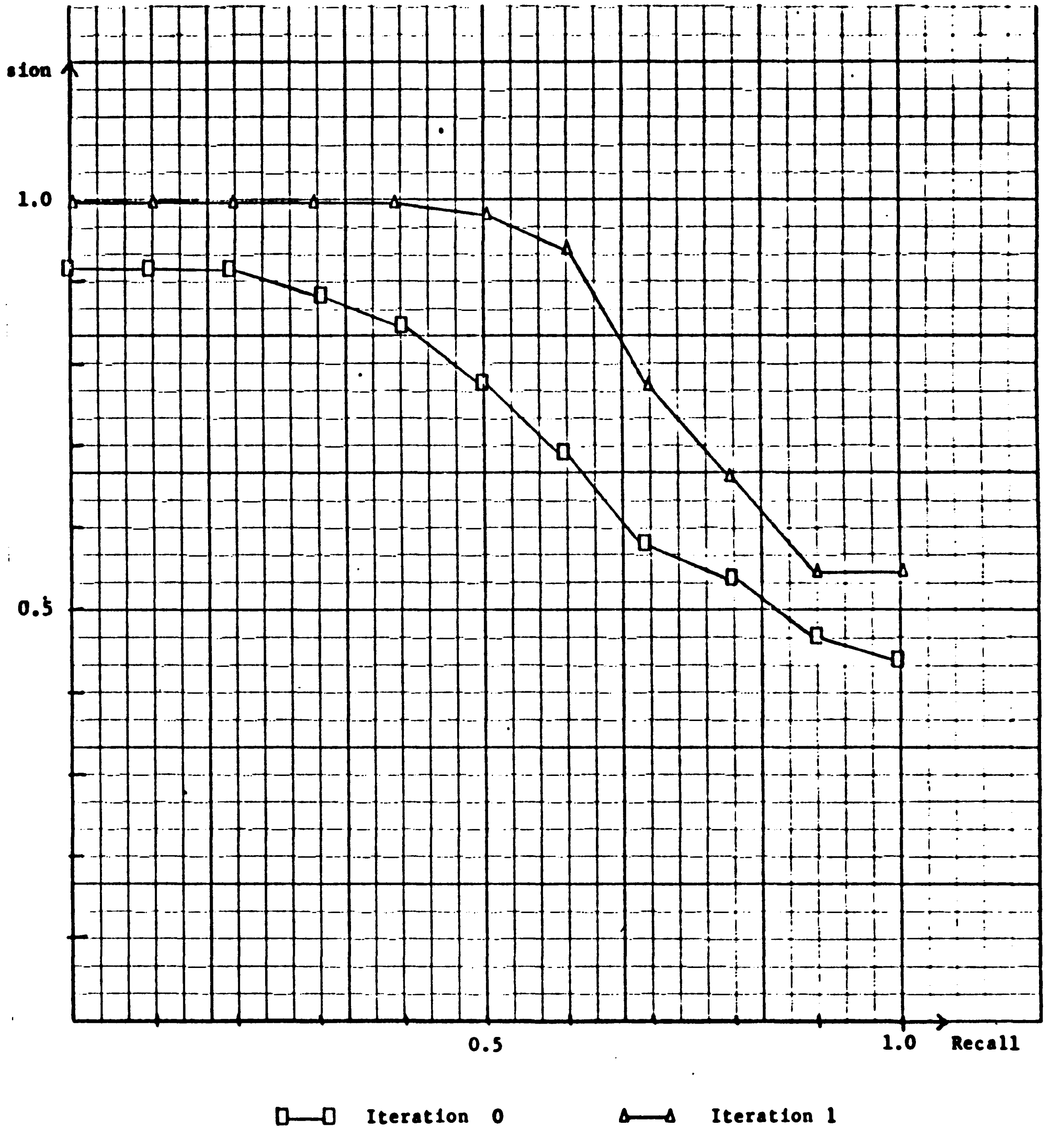


Figure 12: Relevance feedback.



improvement especially at the high recall end of the recall-precision graphs. The results are shown in figure 13.

6.2. Clustering.

While the earlier experiments used a full scan of the collection to evaluate the system performance, this method is clearly not practical in a full scale implementation. It is necessary to group the documents together in clusters. The normal clustering methods from SMART have been used (i.e. Rocchio's algorithm). Initially the clustering experiments were hampered by the fact that it was difficult to find space densities that were high enough to permit sensible clusters. The introduction of backward references helped alleviate this problem. The clusters resulting from clustering runs on the collections with and without backward pointers are shown in figures 14 and 15. Typically the use of pointers helped group letters belonging to the same chain into the same cluster. Whereas for the collection without references the majority of letter chains were broken up over more than one cluster, the use of references reduced the number of chains broken up to 6 out of 41. As the relationship between letters in a chain often results in related documents being relevant to the same query this is clearly a wanted result. Even though no statistically significant improvement was found in the recall-precision graph, it was found that the pointers helped reduce the number of database accesses. A summary of the results is shown in figure 16.

Figure 13: Use of references.

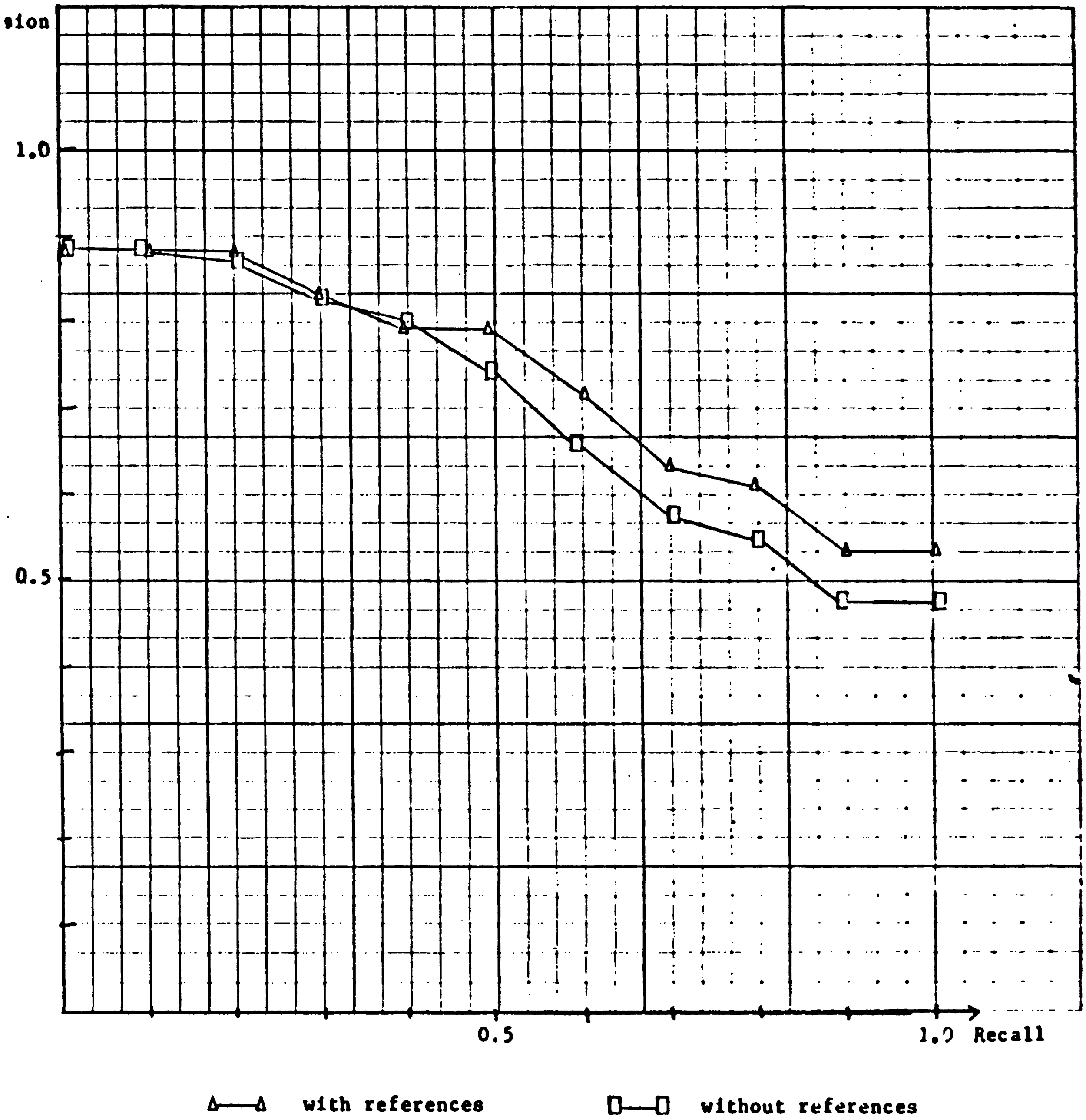


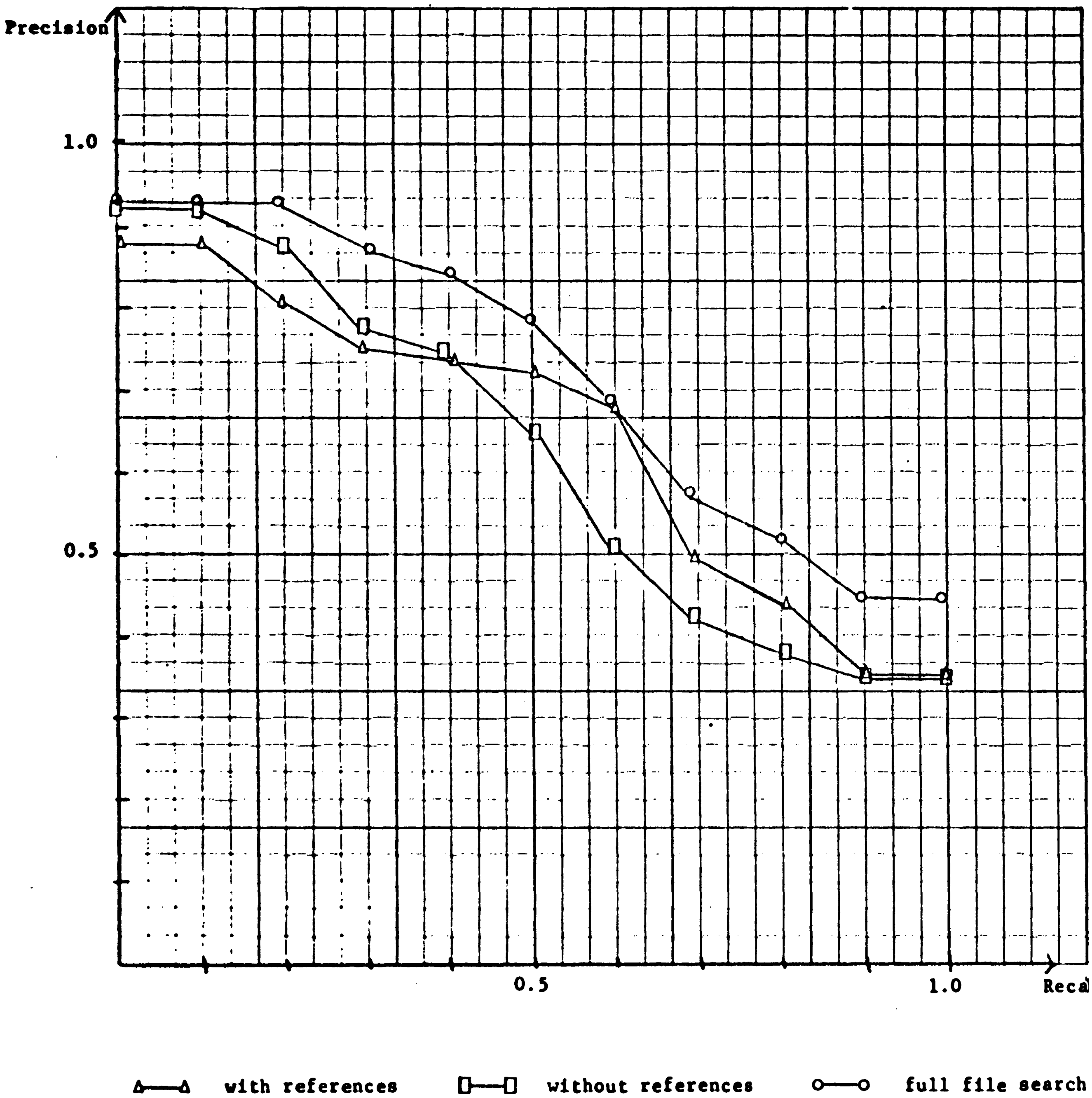
Figure 14: Clusters without backwards references.

Cluster	Documents
1	4 37 38 58
2	6 26 42 48 121
3	8 9 14 87 117 134
4	15 16 17 18 101 102
5	13 20 21 70 91 99 100 126 133 134
6	23 24 27 32 49 51 53 57 64 66 71 80 128 132
7	2 28 50 92 93 127
8	3 29 72 81 82
9	22 33 34 122 129
10	40 61 63 83 88 113 116 118
11	43 45 46
12	43 44 45 46 62
13	59 60 96
14	7 36 65 79 119
15	12 67 68 69 120
16	72 73 74 75
17	35 76 78
18	5 25 84 85 86 115
19	10 31 39 47 52 54 89 90 103 105 109 111 114 125 130 131
20	1 11 30 41 55 56 77 94 95 97 98 104 123
21	19 105 106 107 108 110 112 124

Figure 15: Clustering with backwards references.

Cluster	Documents
1	1 2 41 42 83 128 132
2	5 6 7 10 23 25 26 66 84 85 86 115 133
3	8 9 70 87 134
4	15 16 17 18
5	12 13 20 21 22 99 100 126
6	28 29 50 51 92 93
7	11 33 34 35 36 52 76 77 78 88 97 98 122 123 129
8	4 38 39 40 53 54 113 114
9	43 44 45 46
10	47 48 49 131
11	3 19 55 56 124 125 130
12	37 57 58
13	59 60 95 96 111
14	14 62 63 64 65 94
15	67 68 69
16	72 73 74 75 116
17	24 79 80 81 82
18	89 90 91 127
19	101 102 103
20	104 105 106 107 112
21	108 109 110
22	27 30 31 32 61 71 117 118
23	119 120 121

Figure 16: Cluster search.



The largest problem that occurred during the cluster search was that some of the queries were too specific, wanting letters that had little relationship to other letters. These letters were most often among the documents that had no closely related neighbors, and as a result they were placed in large clusters with little cohesion. A change in strategy of what to do about loose items should help solve this problem.

6.8. Overall evaluation.

The experiments thus far show that the SMART methods are usable in the business environment as well as in the traditional library environment where they were first applied. We therefore claim that the development of a production type retrieval system for business retrieval should be a viable alternative to the existing retrieval methods found in OA-systems.

11

7. Further investigations.

7.1. Introduction.

Effort so far has focussed on development of necessary data files, construction of prototypical tools for preliminary studies, and experiments to validate the general principles of a business correspondence information retrieval system. Further work is possible along several different directions.

In section 7.2 we explore some of the experiments that could be performed using the current data and the current analysis program. They would be relatively inexpensive to do, both in terms of time and effort.

In section 7.3 we explore other types of work that would be feasible. In pursuing these other directions, major revisions to existing data and program files would be necessary, and the resulting cost in terms of time and effort would be much greater than for those changes described in section 7.2.

It is hoped that this chapter, written by Edward A. Fox after discussion with the author and the rest of the Smart group, will be of assistance in strategic planning for further development work on this project.

7.2. Extensions of current work.

7.2.1. Use of letter type in retrieval.

Letter types are discussed in sections 2.2.3 and A.5; it is expected that a final list of types relevant to the existing document collection could be quickly developed. Given such a list, a human "judge" could classify each document as to its type (or types), and, simultaneously, record those phrases which were influential in the decision process. The resulting list of phrases could be used later to prepare a "feature vector" based automatic routine for classifying documents as to type. For further discussion, see section 3.2.4.

Given letter types for all documents, several experiments could easily be performed. The clustering work described in section 6.8 could be extended by re-clustering the collection, first using content terms plus type (with a suitable weight chosen for each of the applicable types) and secondly using content terms plus back references plus type. Then, existing query collections could be used for searches, so as to allow recall-precision comparisons against searches made of files lacking type indicators (see figures 14-16).

A further utilization of type would be in a query enhancement procedure where terms were automatically added to queries based on an analysis of the original query. Thus, for example, a query with term "unfair" might have the letter type "complaint" automatically added in. Such an automatic process could be manually simulated, using the phrase list constructed during the manual type-classification activity (see above). Once again, recall-precision comparisons could be made to validate the effectiveness of such query enhancement procedures.

7.2.2. Use of factual identifiers.

Business letters contain both content-terms and factual identifiers (ex. date, addressee, author). Preliminary experiments have begun as to the value of such factual identifiers; see the discussion at the end of section 6.1 and that in 6.2.2.

One experiment, that should be further developed, was devised to compare two promising storage and retrieval methodologies. In one, the standard approach, an inverted file was created utilizing all content and factual terms. In the other, one inverted file was created for factual terms, and a second inverted file was devised to refer to the low level (content term) cluster centroids. To search for relevant documents, the inverted files were first referenced. The second storage method, however, required an additional step of comparison and ranking of cluster centroids, which, it was hoped, would accurately rank the output set of retrieved documents.

This experiment is still in progress and will require additional investigation. Certain parameters, like cluster and page size, should be varied to better model real-life behavior. Different clustering runs should be made, to determine the benefits of including backward references and letter type in the list of content terms. Finally, a set of longer queries, hopefully with more relevant documents per query, should be selected, so the output ranking method can be better judged.

Other experiments are also possible to explore the effect of factual identifiers. Certainly, as in section 7.2.1 above, such data elements as addressee and date could be included in the list of content terms. Re-clustering could then be done, to see if such elements were of value in improving recall-precision performance. Weighting schemes for factual identifiers could also be explored, to see if such terms behave differently from the normal content terms.

7.2.3. Use of simple business thesaurus.

Originally, a business thesaurus was called for in this study; see sections 2.2.5 and 3.2.6-8. Much of the preparation for this has been done: phrases are given in appendix A.1, abbreviations in A.2, thesaurus classes in A.3, and synonym lists in A.4.

Given this information, the letter collection could be edited and re-indexed so as to include thesaurus classes and phrases in a new indexing dictionary. Searches could then be made adding phrases only, adding thesaurus classes only, or adding both phrases and thesaurus classes. These results could then be compared against previous results to validate the expected performance improvements.

7.3. Further work.

7.3.1. Automatic tools.

The experiments outlined in section 7.2 could all be done with existing facilities without extensive programming, in order to validate the ideas suggested. If results so warrant, automatic programs could be developed to facilitate future work in these areas.

In particular, the determination of letter type and the construction of a business thesaurus may be somewhat automated. Given a list of letter types, phrases (that statistically are good indicators of type) could be selected automatically; a human judge could then edit them for semantic appropriateness. Using the letter types and edited list of indicator phrases, experiments could then be conducted to select the best approach to automatic letter-type determination.

Similarly, to construct an appropriate dictionary, statistical methods can be utilized to isolate some phrases, based on co-occurrence information. Statistics could also help group items into thesaurus classes, using content identifiers that co-occur in very similar documents or that satisfy some partial matching condition.

Some of these tools should be incorporated into an improved letter analysis program; others should be part of a (prior-to-indexing) statistical package of general applicability to the Smart system.

7.3.2. Bigger collection.

The current collection includes 134 letters and two query sets, one of size 25 and the other of size 19. To give more conclusive results, and to allow more effective clustering, a larger collection from the existing file of several thousand letters, namely some 400 documents, might be entered. Several other query sets could also be solicited, and relevance judgements obtained, so that a query collection of size 40 or 50 could be used for experiments. Such larger query collections should also be obtained to satisfy the input requirements of experiments with combined content and factual data components.

7.3.3. Other collections.

The current collection is of the "custom" type; it would be interesting to work with a "production" type of collection as well. Of course, then, new queries would be needed. Probably, the letter analysis program would have to be revised or generalized. Hopefully, automatic methods (see 7.3.1 above) could be used to rapidly develop a list of letter types and indicator phrases as well as to construct a business thesaurus; in any case, some manual revisions would surely be necessary. All in all, then, such development of other collections, while of undoubted practical importance, would certainly require a good deal of effort.

7.3.4. Packaging.

In order to implement a usable information retrieval system for business correspondence, all of the components of this project, as well as those of the Smart system needed for clustering and searching, would have to be generalized and appropriately packaged. Such an activity would require a fair amount of effort, and is beyond the scope of this investigation.

7.3.5. Other investigations.

Finally, there are many new ideas related to combined retrieval methods that could be analyzed in the context of existing business correspondence collections. Such work is, however, beyond the scope of this initial study.

REFERENCES

Bair, J.H., "Avoiding Working Non-solutions to Office Communications System Design", COMPCON, Spring 1980, Computer Society of the IEEE, San Francisco, 1980

Bayer, R. & McCreight, E.,"Organization and Maintenance of Large Ordered Indexes", Acta Informatica, Vol 1, p 173-189, 1972

Cresci, M.W., "Complete Book of Model Business Letters", Parker Publishing Company Inc., West Nyack, New York 1976

Diebold Automated Office Program, "An overview of Office Automation Tools", Diebold Inc., 1979

Ellis, C.A. & Nutt, C.J., "Office Information Systems and Computer Science", Computing Surveys, Vol 12, No 1, March 1980

Knauss, R. "Pattern Based Semantic Decision Making, Empirical Semantics", Burghard Rieger ed.,Brockmayer, Bochum, West Germany, 1980

Lodahl, T., "Management Support Systems in the Complex Office", Office Systems Research Group, Cornell U, Ithaca, 1979

Lovins,J., "Development of a stemming algorithm", Mechanical Translation and Computational Linguistics, Vol. 11, p 22-31, 1968

Maedke, Robak & Brown, "Information & Records Management", Glencoe Press, 1974

Morgan, H.L., "Office Automation Project - A Research Perspective", Proc. AFIPS 1976 National Computer Conference, Vol 45, AFIPS Press, Arlington, Virginia, pp 605 - 610

Palme, J., "Human Needs and Office Systems", Proc. Stanford University Symposium on Office Automation, March 26-28, 1980

Paice, C. D., "The Automatic Ceneration of Literature Abstracts: an Approach Based on the Identification of Self-indicating Phrases", BCS-ACM Symposium on Research and Development on Information Retrieval, Cambridge, England, June 1980

Purchase, A. & Clover, C., "Office of the Future", SRI Long Range Planning Service, Number 1001, April 1976

Rogers, E.M., "Once a System is Invented, How Do We Cet It into the User's Hands?", Proc. Stanford University Symposium on Office Automation, March 26-28, 1980

Rothnie et. al.: "Introduction to a System for Distributed Databases (SDD-1)", ACM Transaction on Database Systems, Vol. 5, No. 1, p 1-17, March 1980

Rush, J.E., Salvador, R. & Zamora, A.: "Automatic Abstracting and Indexing. II. Production of Indicative Abstracts by Application of Contextual Interference and Syntactic Coherence Criteria", Journal of ASIS, Vol 22, No 4, p 260-274, 1971

Salton, C., "The SMART Retrieval System, Experiments in Automatic Document Processing", Prentice Hall, Englewood Cliffs, NJ, 1971

Salton, G., "A New Comparison Between Conventional Indexing (Medlars) and Automatic Text Processing (SMART)", Journal of ASIS, Vol. 23, No 2, pp 75-84, March 1972

Salton, C., "A Theory of Indexing", Regional Conference Series in Applied Mathematics, Vol 18, Society for Industrial and Applied Mathematics, Philadelphia, 1975

Van Rijsbergen, C.J., "Information Retrieval", (1.ed.), Butterworths, London, 1975

Vaskevitch, D.: " The Evolution of the Office Automation Marketplace", NCC 1980 Office Automation Conference Digest, AFIPS press, Atlanta Georgia, 1980

APPENDIX A: BUSINESS THESAURUS

OUTLINE

A.1. Index phrases.

automated document atorage and retrieval

automatic abstracting

automatic data retrieval

automatic document proceasing

automatic document retrieval

automatic feedback

automatic indexing

automatic language processing

automatic text analysis

automatic text processing

automatic thesaurus construction

B-tree

back-of-the-book indexes

clustered files

computational linguistics

computer conferencing

computer network

computer science

content analysis

content identification

data base

database retrieval

directory of courses

discrimination value

document collection

document frequency

document retrieval

dynamic collection control

dynamic library

dynamic library management

fact retrieval

feedback retrieval

file clustering

fuzzy matching

graduate students

information retrieval

information science

information storage and retrieval

information systems

inverted files

lecture notes

letter-to-the-editor

library automation

library of the future

man-machine interaction

on-line retrieval

one-pass

operating systems

programming techniques

proper stem

pseudo-classification

question-answering systems

reference retrieval

relevance assessments

relevance feedback

research assistant

sabbatic leave

signal-noise

SMART-Medlars comparisons

state-of-the-art

stemming procedures

storage organization

subject retrieval

synonym dictionaries

term classification

text analysis

text processing

user interface

user's manual

vector space

word discrimination

A.2. Abbreviations

American Association for the Advancement of Science	AAAS
American Journal for Computational Linguistics	AJCL
American Society for Computational Linguistics	ACL
American Society for Information Science	ASIS
Annual Review of Information Science and Technology	ARIST
Artificial Intelligence	AI
Association for Computing Machinery	ACM
British Library, Lending	BLL
Communications of the ACM	CACM
Computing Reviews	CR
Computing Surveys	CS
Conference Board of Mathematical Sciences	CBMS
Council for Library Resources	CLR
Fall Joint Computer Conference	FJCC
Information Storage and Retrieval	IS&R
International Forum for Information Science	IFIP
Journal of ASIS	JASIS
Knowledge Representation Language	KRL
National Bureau of Standards	NBS
National Computer Conference	NCC
National Institute of Health	NIH
National Library of Medicine	NLM
Society for Industrial and Applied Mathematics	SIAM
Special Interest Croup, Information Retrieval	SIGIR
Special Interest Croup, Office Automation	SIGOA
Spring Joint Computer Conference	SJCC
Transactions on Database Systems	TODS

A.3. Classes.

The functional classes are based on occurrences found in the test collection.

Conferences, meetings:

FJCC, SJCC, NCC, ASIS annual meeting/conference, IFIP congress, AAAS annual meeting/conference, SIGIR session, Regional Conference on Automatic Information Organization and Retrieval

Journals:

Computers and Automation, CACM, JASIS, IS&R, Information Processing Letters, TODS, Kibernetiky Sbornik, JACL, CR, CS, SICIR Forum

Former Ph.D. students:

Bergmark, Dattola, Rocchio, Williamson, Wong, Yang, Yu

Geographical:

National:

Ithaca, Syracuse, L.A., San Fransisco, Denver, Rochester

International:

Abroad

Europe:

Europe, France, Switzerland, Munich, Zurich

Other:

Canada, Australia, Mexico, Mexico City

Publishers:

McGraw Hill, Prentice Hall

Professional Associations:

ASIS, ACM, ACL, SIAM, AAAS

Operational Retrieval Systems:

Responsa, SIRE, DIALOG, OVID, SMART, RADCOL,
New York Times Information Bank

Universities:

Cornell, Harvard, Stanford, University of Missouri, Bar-Ilan
Weizmann Institute

Business firms:

Xerox, Systems Development Corporation, New York Times,
IBM, Lockheed

Information Retrieval Researchers:

Fairthorne, Bar-Hillel, Lancaster, Mooers, Fraenkel

Programming Languages:

APL, BASIC, COBOL, FORTRAN, PL/1, SNOBOL

Hardware:

IBM/370-158, 360-65, IBM 360/65, IBM 7094, IBM 360-67

A.4. List of synonyms

abstract - summary

accesible - available

advocate - recommend

Annual Review - ARIST

assist - aid - help

associate - connect

attempt - try

bibliographic information - references

book - monograph

Communications - CACM

directions - instructions

discuss - talk

enclose - attach - inclose

entitle - title - name

factual - bibliographic - non content

findings - results

fishy - dubious

form - create - build

funds - money

generate - produce

handouts - materials

imply - indicate

indicate - show

interactive - on-line

keylist - vocabulary

lecture notes - syllabus

lecture series - course

manuscript - sheets

option - alternative

paper - publication - article

preface - summary - abstract

previous - earlier

procedure - algorithm - routine - program

professor - faculty

read - study

regret - sorry

reprint - copy

run - operate

seminar - colloquium

shortcomings - flaw - errors

stipend - grant - fee

subsume - include

symposium - conference - meeting

teach - lecture

term - concept - word - word stem - stem

topic - theme

train - instruct

transmit - send

unable - cannot

understand - comprehend

visit - call on - come - stop by

work - efforts - research

A.5. Types

Presently only a very preliminary list of type indcating phrases exist. It will, hopefully, be printed with reports yet to come.

A.6. References

attach

enclose

inclose

letter dated ..

letter of ..

note dated ..

note of ..

recent letter

recent note

A.7. Salutations & Closes

a. Salutations

Dear ..

Madam

Sir

To whom it may concern

[start text]

b. Closes

Best wishes

Best regards

Cordially [yours]

Faithfully [yours]

Sincerely [yours]

Truly yours

Very truly yours

Yours cordially

Yours faithfully

Yours sincerely

Yours truly

Yours very truly

A.8. Courtesies

Entries ending with a '.' symbolizes truncation. (i.e. all strings matching up to the period is accepted as succesful matches).

all good wishes

all the best

all your help

anticipat.

any assistance you

any help you

appreciate any

appreciate your

appreciat.

assist

at your service

best regards

best wishes

children

delight.

disappoint.

Easter

enjoy

everybody are

expect to write [to you] more fully later

family

favorable consideration

feel free

fight the good fight

food

for your consideration

for your success

fun

further help

give my regards

good luck

grateful

Happy New Year

hearing from you

help

hockey

holiday

hope that all

hope thst this

hope business is good

hope tó see you

hoping to hear from you soon

how do you

how can I assist you further

husband

in anticipation of your reply

in this respect

keep in touch

kind.

look forward to

looking forward to

many thanks

many thanks again

may we have the pleasure of hearing from you

most thankful

Merry Christmas

New Year

once again

personal regard.

pleasant

please

please be so good

please don't let us down

please give this

please let us hear from you

please let us know

pleased to

pleasure

rain

regards

regret

remember me to

rest assured

say hello to

send my regards

shall be glad to hear from

ski

skiing

skis

snow

sorry

spouse

take it to heart

take pleasure

take the liberty

thank you

thank you again

thank you for letting us know

thank you in advance

thank you once again

thank you very much

thanking you for

thank.

trusting this meets with your approval

wife

with all good wishes

your early attention will be appreciated

your point is