Evaluation of Text Indexing and Retrieval in the Office Environment

W. Bruce Croft
James Rochfort

COINS Technical Report 81-27

December 1981

This research was supported by the Digital Equipment Corporation.

1. Introduction

The advent of office information systems consisting of personal workstations linked into a local network will dramatically change the operation of a business. Much of the work on these systems has concentrated on the development of editing, production and mailing facilities for text documents. There will be a number of types of text documents in the office environment, including reports of various lengths, formal memos, letters and casual messages. A large number of these documents will need to be filed and possibly retrieved at a later date. Surprisingly, little emphasis has been given to techniques capable of performing these tasks effectively. Instead, users have been forced to remember information such as floppy disk numbers in order to find the relevant documents. It should be an important aim of the designers of office information systems to provide better filing and retrieval methods than any manual system.

This report describes a series of experiments designed to evaluate the effectiveness of text indexing and retrieval techniques in the office environment. Although many experiments have been carried out with bibliographic databases containing scientific journal articles, some techniques may require modification to allow for the characteristics of business text and to make full use of the features of a word processing/personal workstation environment. To provide a basis for the discussion of the experiments, we shall introduce indexing and retrieval techniques as they have been applied to previous research. The most appropriate techniques for business text will be suggested and, in the cases where there is not

enough information available, experiments designed to provide this information will be described. In the last section, we will summarize the results of the experiments and describe a possible design for a text filing and retrieval system.

1.1 Indexing Techniques

In this section we will discuss techniques which are used to represent the content of documents using sets of keywords or index terms. The following assumptions are made

(a) It is desirable that the index terms are derived automatically or at least semi-automatically.
(b) Each document will be input according to some standard form and therefore "fixed" keys such as source, destination and date for a memo can easily be identified.
(c) Efficiency considerations are important.
(d) Indexing will be based on the entire text.

Assumption (a) is made on the grounds that professional indexers will not generally be available in the office environment and that manual indexing by individual workstation users would be inconsistent and its use would lead to poor retrieval performance in many cases. Assumption (b) means that certain fixed characteristics of the documents will always be available for use in retrieval. Assumption (c) implies that effectiveness is not the sole criterion for judging a technique. We use effectiveness here to mean the

ability of the system to separate the relevant and non-relevant documents for particular queries. Efficiency is an important consideration because these techniques will be required to work on workstations with less processing power and storage capacity than large mainframes. Fortunately, it has been shown that the simplest and most efficient techniques are often the most effective. Assumption (d) is made because the experiment uses a variety of documents from the office environment rather than scientific documents. Experiments have shown that, for scientific documents, satisfactory performance is obtained using the title and abstract rather than the full text. Indexing the full text is obviously less efficient. However, business documents do not, in general, contain abstracts. This is not a problem with memos because they tend to be short and, therefore, indexing the full text is practical. However, indexing longer, unstructured documents could be difficult without using the full text. The efficiency of using full text is the subject of one of the experiments reported here. Another solution would be to index the title and section headings (if any) and have the user supply extra terms if necessary. A similar method could be used to index and file documents which are not in machine-readable form.

On the basis of these assumptions, we shall be concentrating on efficient automatic indexing techniques which use the full text of the document. The major sources of information for this section are Salton [1], Van Rijsbergen [2] and, in particular, Sparck Jones [3,4]. These references describe a large number of experiments which used indexing techniques on a variety of collections of scientific documents. The features of the METER system [5] will

also be used for comparison where it is appropriate.

The indexing process produces keywords by a statistical analysis of the occurrence of words in text. Prior to this analysis, the word forms are usually normalized by a suffix stripping procedure and perhaps by identifying synonyms. More complicated procedures, even those introducing simple concepts like phrases, have not been shown to be more effective than indexing using word stem occurrences.

The first step in the statistical treatment of the text is to remove any non-alphabetic characters and, usually, to remove words of less than 3 characters in length. However, in an office environment, there will probably be significant 2-letter acronyms and it would be detrimental to retrieval performance to remove these. This leads to

Experiment I1: In the given sample of text, calculate the number of useful two-letter words or acronyms as a fraction of the total number of useful words.

If a significant number of 2 letter words do exist in business text, a procedure for handling them will have to be developed.

A similar concern arises with peoples' names. These are not important in scientific text but could be very important in business text. They can of course be treated as normal index terms but it may be desirable to link name occurrences in the text with similar names used in the source and destination fixed keys. This would require special procedures and, to decide if the effort is justified, we have

Experiment I2: In the given sample of text, calculate the number of

names as a fraction of the total useful words.

The next step in the processing is to remove common words (such as 'and', 'or', 'the') by consulting a "stopword" dictionary. In previous experiments, these dictionaries have been fairly small (approximately 200-300 words). In a word processing environment, the text is often checked for spelling errors by a program. These programs typically use a large dictionary of words. It may be possible, therefore, to use this spelling dictionary for a number of purposes in the indexing and retrieval system. One possibility is to increase the number of stopwords by specifying this property in the dictionary. This may require someone in a particular office to designate stopwords, but this should be no more difficult than adding new words to the dictionary which is done in any case. To see how many extra words may be designated as stopwords, we have

Experiment I3: In the given sample of documents, calculate the number of words (after stopword removal) that may also be considered stopwords.

Other uses of the spelling dictionary will be mentioned later.

The next step is to reduce the words to stems by suffix stripping. This is important both to reduce the size of the final vocabulary and to identify similar words which are in different forms. The METER system claims to have a sophisticated stemmer and it may be useful to have this for future work. However, the Cambridge stemming algorithm which will be used in our experiments has been shown to be as effective as many other algorithms and it has the additional benefit that it is much more efficient than most

algorithms.

After stemming, the number of occurrences of each stem in the individual documents are totalled and document representatives consisting of sets of index terms (the stems) with frequency counts are produced. At this stage, a number of statistics can be gathered.

Experiment I4: Calculate the total vocabulary size (the number of unique word stems). Calculate the vocabulary size when extra stopwords identified in I3 are removed.

Experiment I5: Calculate the average document representative length (number of stems). Calculate the average length with extra stopwords removed.

Experiment I6: Calculate the distribution of frequency counts in the documents.

These statistics are valuable for the following reasons. The total vocabulary size indicates the amount of index overhead required. The average document description length also affects the storage overhead. Sparck Jones [4] has shown that document length is not really important for retrieval effectiveness, but she has also shown that it is detrimental to delete terms from the vocabulary in order to reduce the overhead, even if sophisticated methods are used. Therefore, we do not recommend the heuristic term deletion method used in METER. One method Sparck Jones mentions for reducing the size of document desciptions without affecting performance is to delete terms with frequency 1 in individual documents. Experiment I6 will indicate how many terms can be removed by this method. This experiment will also show if search

strategies based on within-document frequencies will have enough information (a broad range of frequencies). On the basis of this experiment, binary document representatives, in which terms are considered to be assigned (weight 1) or not assigned (weight 0), may be recommended.

The final step in this indexing process is to construct a stem dictionary to replace stems with stem numbers in the document representatives.

Two experiments dealing with relationships between index terms are

Experiment I7: Calculate the number of abbreviations or acronyms present in the final vocabulary.

Experiment I8: Examine statistically similar pairs of stems to see if they are reasonable synonyms.

Experiment I7 will give some indication of the frequency of synonyms in the form of abbreviations or acronyms. If they are common, some form of thesaurus will be a necessary part of the system. Experiment I8 will use a term classification method [4] to generate groups of index terms. These terms can then be examined to see if they would be useful for user browsing and query modification.

1.2 Retrieval Process

The experiments on the indexing process cannot be completely separated from the retrieval process because a change in the indexing will affect the retrieval performance. Retrieval

experiments will be required to back up some investigations in the previous section as well as to compare different retrieval strategies. The techniques discussed in this section are based on work in Van Rijsbergen [2], Harper[6], Croft [7], Croft and Harper [8] and Sparck Jones [9].

To perform these experiments, we will need a user or group of users to provide a sample set of queries and to look through the retrieved documents and the document collection to identify relevant documents. The evaluation methods used are based on two measures, Recall and Precision. Recall is the proportion of the total relevant documents for a query that were retrieved. Precision is the proportion of the retrieved documents that were relevant. Precision can be measured just by looking at the retrieved documents but in order to measure recall accurately, the entire document collection must be examined for relevant documents. The identification of relevant documents will require the full text of the documents.

The main series of experiments will be as follows,

Experiment R1: Verify that search strategies based on probabilistic models are more effective than other strategies with business text. This experiment requires textual statements of interest for the queries.

Experiment R2: Compare the performance of Boolean query specifications to the performance obtained using natural language queries and the probabilistic model.

Experiment R1 will use the best known retrieval strategies on the document representatives derived in the indexing process. The

obvious approach to retrieval using index terms is simply to rank documents in order of the number of terms they have in common with the query. This assumes that the natural language query is indexed in the same way as the documents. This retrieval method (coordination matching) can be regarded as the base line of performance. A more sophisticated method called "relevance weighting" is based on a probabilistic model of retrieval. A weight is calculated for each term based on its occurrence in the sets of relevant and non-relevant documents for a given query. Initially, of course, the only information about the relevant set of documents for a query comes from the query itself. The term weights are then totalled for terms occurring in individual documents and the documents are ranked in order of the total score. This process is equivalent to ranking the documents in order of their probability of being relevant to the given query and it has been shown in many experiments to be very effective. Experiment R1, therefore, will be used to confirm that the sophisticated strategies are more effective than simple coordination matching for business text. It will also give some indication of the absolute performance level obtained using these techniques.

Experiment R2 involves user interface issues in that it investigates the relative performance of Boolean queries compared to natural language queries. This experiment will require the user to reformulate the natural language queries in terms of Boolean expressions.

An important aspect of searching that will be studied here is relevance feedback. In this process, the user identifies relevant documents that have been retrieved in the initial pass of a search

strategy. The system then uses this information about relevance to find more relevant documents by using relevance weighting. Experiments with relevance feedback require no more information than the relevance judgements mentioned previously. If the users are not prepared to do exhaustive relevance judgements, the experiment can still be carried out in the following way. The top 10 (say) documents from the initial ranking are presented to the user. The user then identifies relevant documents in this group and the relevance feedback process is invoked. The top 10 documents (excluding those already seen) from the new ranking are then shown to the user for relevance judgements. This evaluation process will at least provide some information about the usefulness of relevance feedback in this environment.

Experiment R3: Investigate the effectiveness of relevance feedback in the business environment.

Relevance feedback may be particularly useful when the user is looking for a very specific document or group of documents. This situation will probably occur more often in a business environment than in a general bibliographic retrieval system. Therefore, queries which are very specific should be identified and the following experiment performed.

Experiment R4: Investigate strategies for answering very specific queries.

Many of these more specific queries may turn out to be more appropriately specified with a Boolean formulation - or at least a Boolean expression could be used to specify the subset of documents to be searched in more detail.

One further topic that should be mentioned is the use of clustering techniques to group documents. These techniques are similar to those used to group related index terms for browsing. Clustering methods tend to be expensive, but some recent experiments have indicated ways in which this process can be made more efficient. Clusters can be used by the system in various search strategies designed to improve the effectiveness of retrieval. Because they are generated statistically, the clusters generated need not represent a classification that makes sense to the users of the system. In fact, the most useful types of clusters for retrieval are very small, consisting of 2-5 documents. For classification purposes, it may be better to have the user describe different categories and the system would compare incoming documents to these categories, classify them and present them to the user as ranked lists within categories. This would provide a filter mechanism for people who are sent a large number of messages. The investigation of a filtering mechanism will not be pursued in the current proposal.

A clustering algorithm would perhaps provide assistance to the user in the description of categories of interest if only a few large clusters were generated. It would be useful, therefore, to study the type of document groups generated by automatic clustering algorithms.

Experiment R5: Examine document groups generated by clustering algorithms.

Another possible way of specifying categories of interest is for the user to give examples of documents in each category. The system would then be able to derive the features of the categories from

these documents.

2. Experimental Results

2.1 Constructing the test collection of documents

A decision was made to obtain two types of databases, one containing documents generated on an electronic mail system and the other containing memos, reports and other types of business correspondence. The experiments performed on these databases would then highlight the differences between them. This ideal separation of formal and semi-formal business documents does not, of course, happen in practice but we feel that we have obtained a reasonable distribution of documents. The sources used were two mail systems (M1 and M2) and a personal file of reports for word processing using a formatter. The M1 system tends to contain more informal messages than the M2 system. The sizes of the databases are fairly small, but this does permit a more thorough judgement of relevance to queries. To summarize, the databases being used are

Collection	Type of document	No.of Documents
D	M1, M2 combined	250
H	M2	75
G1	M1	90
G2	formatter files	40

The headers of the mail documents have been stripped off and

placed in separate files. The standard form of document produced by the programs consists of a unique identifier followed by the text and a special document terminator marker. The next step in the document processing removes stopwords, two-letter words, non-alphabetic characters and suffixes. Eventually the standard document consists of the unique identifier followed by a set of pairs of numbers and the terminator marker. The first number in the pair refers to a term or keyword in the dictionary and the second is the frequency of that term in the document. The statistics from the first indexing step are as follows,

Collection	D	H	G1	G2
Words in docs.	17138	16003	17837	29468
Stopwords	4182	4065	4387	10370
Words output	8568	8234	9248	16391
Average per doc.	68.6	213.4	198.2	736.7

These statistics give an indication of the type of documents in each collection. The documents in the D collection tend to be much smaller than those in the other collections (3-4 lines average). The documents in the formatter collection (G2) are much longer (3-4 pages average).

2.2 Two-letter Words (Experiment I1)

The document processing routines currently remove all two-letter words from the text. There is strong possibility that

some of these words will be important acronyms for a particular company. In order to get an idea of the frequency of important acronyms, the two-letter words in the entire sample of documents were examined. Appendix 1 gives a listing of the results. It appears that there are some important acronyms used in the mail system documents, but their relative frequency is very low (approximately 15 in 415 documents). This means that they could be handled efficiently with a simple table lookup.

2.3 Peoples' names (I2)

The number of names occurring in the various collections expressed as a percentage of the number of unique stems appears below.

Collection	Percentage of Names
D	11.3
H	15.8
G1	8.8
G2	11.0

These percentages were obtained by manual inspection of the stem dictionaries. The experiment shows that peoples' names are indeed an important component of business text. However, many of the queries concerning names will probably deal with the fixed field information such as source and destination. For example, 'List all documents written by Bates last month.' The names which were

examined in this experiment will be available as normal content search terms. An example of the type of query using these names would be 'List the documents which mention Fred Bates.'

One design decision that will affect the size of the total index vocabulary is whether to retain first names or to include them in the stopword dictionary and remove them. This decision depends on the importance of first names for identifying documents. That is, if the system contains a large amount of casual mail that must be indexed, then first names may be the only names mentioned. In this case, they should be retained as index terms.

2.4 Spelling Errors

It bacame evident while examining the stem dictionaries that many words were misspelt. This misspellings would be greatly reduced by the use of a spelling checker, but for this experiment the percentages of errors were calculated and they were left in the vocabulary. This may have a detrimental effect on retrieval performance.

Collection	Percentage Spelling Errors
D	9.2
H	2.6
G1	2.1
G2	2.3

2.5 Extra Stopwords (I3)

The stopword list used in processing the documents consisted of 250 words. The stem dictionaries produced were then examined manually to determine the number of words still present that may be considered stopwords. The figures for the various collections are as follows,

Collection	Percentage Extra Stopwords
D	6.1
H	4.9
G1	6.8
G2	5.5

The most common of these extra stopwords were combined with the old stopword list to form a new list of 288 words.

The experiment indicates that provision for the addition of extra stopwords should be made, but the total stopword list size will remain small.

2.6 Collection Statistics (I4,I5)

Statistics for both the documents and terms for all collections appear in Appendix 2. From these statistics, we can make the following statements

1. The G2 collection contains a much higher proportion of long documents than the other collections.
2. The D collection contains more small documents than the other collections.
3. All collections except G2 contain a few documents with almost no content (less than 3 index terms). Collection D contains the most of these documents, but they are still a small percentage of the total.
4. All collections contain some very large documents (>200 index terms).
5. Removing stopwords and short words reduces the documents to about one-half of their original length. Stemming reduces the number of unique words remaining by 15% for the D collection through to 56% for the G2 collection.
6. The removal of extra stopwords does not sharply reduce the size of the indexed documents. Size reductions of between 5 and 10 percent were obtained.
7. The vocabulary size is about the same for all collections (approx. 2000 terms). Well over 50% of these terms occurred in only one document. This can be attributed to the small size of the sample collections of documents. If the sample size were increased, we could expect a decrease in the proportion of terms posted once and a slow increase in the vocabulary size.
8. The extra stopwords that were removed contained both frequently posted terms and terms used once.

The overall conclusion would be that while many of the indexed documents contain reasonable numbers of terms and can be stored

efficiently, many are still too long.

2.7 Thresholding and distribution of frequency counts (I6)

Appendix 3 lists the distributions of frequency counts for the terms in all collections. This data shows that although most of the terms occur with frequency 1 in a document, a large number do occur frequently within documents. This would be particularly true for longer documents. The G2 collection which has a similar number of postings to the D collection but contains longer documents, shows a much larger proportion of high frequency counts. Therefore, it is suggested that terms of low frequency be deleted in longer documents in order to reduce their size. Appendix 4 shows the figures for the collections after terms of weight 1 have been deleted. These should be compared to the document statistics in Appendix 2 (after removal of extra stopwords). It is clear that deletion of low weight terms is very effective in reducing the size of longer documents. For example, the average length of a G2 document dropped from 508 to 34 after the deletion of terms of weight 2 or less. This deletion strategy should only be applied to longer documents - short documents may have only terms of low weight assigned. The deletion of low weight terms will also have the effect of reducing the total vocabulary size.

2.8 Abbreviations and acronyms (I7)

The following table lists the percentages of abbreviations and acronyms in the collections (excluding 2 letter acronyms mentioned in I1).

Collection	Percentage Abbrev.
D	7.3
H	3.8
G1	7.8
G2	6.3

These figures indicate that a significant number of terms in the vocabulary are indeed synonyms of other words. These relationships can be indicated in the spelling dictionary.

2.9 Statistical synonyms (I8)

For this experiment and the document clustering experiment mentioned later, a classification procedure is used that is similar to the string generators mentioned in [4]. Strings are a connected series of nearest neighbours and they can be generated efficiently using the inverted file. The main problem which arose with term clustering is that it depends on the cooccurrence of terms in documents but the collection used (H) was too small to provide significant information. Therefore no useful term groups were

produced.

2.10 Retrieval Experiments (R1-R4)

The results of these experiments were limited due to a lack of time and the lack of direct participation by the users involved with the project. However, in general the results show that standard retrieval techniques developed for scientific text are also appropriate for business text. The search strategy used was the simple match - the documents were ranked according to the number of matching terms with the query (the relevance weight). A number of other techniques have been shown to be more effective than this one, so the results can be regarded as a baseline performance.

The query collections used in the experiments were considered adequate as they contained:

1) Queries for specific documents.
2) Queries for general classes of documents.
3) Queries requesting documents not in the data base.
4) Queries slightly off the issue - i.e. queries that were 'almost' correct, but were a little bit confused.

In the first category, the correct document was assigned the highest weight in every case. The retrieval was correct the first time, every time, with a relevance weight far above documents that did not answer the query exactly. For example, the query "find the memo re: videotape on ems for Bruce Steward" has keywords of

'videotape','ems','Bruce',and 'Steward'. Those memos regarding videotapes about ems had at most a base weight of 2 (videotape and ems) where the actual answer had a base weight of 4 (all keywords). Most queries fell in this category (note that this category overlaps with categories three and four); a total of eight queries.

In the second case, the relevant documents were clustered right at the top of the list. There were not, however, queries relating to groups of documents so large as to overflow the top ten of the 'retrieved' list, and inter-document relevance was not judged; it was considered sufficient that all documents successfully answered the query. An example here is the query "Is security on the mail system a problem?". Note that in fact this query is beyond the scope of the system we originally intended to test, i.e. it is not requesting the retrieval of a document, rather it is asking a general question. The anwer to the question was however contained in the retrieved memos, all of which ended up with a weight of at least 3 out of a maximum of 4.

In the third case, the results were encouraging because no document was ever retrieved with any significant weight. It would then appear that a 'threshold' of relevance could be implemented easily - i.e. instead of a group of low weight documents, the simple message "No Relevant Documents Were Found" would appear to the user. The exception to this is the query "Who is Jane Gizzonio?". There was one memo that asked for the person to please call Jane Gizzonio, but with no reference to who she was. It is arguable that this retrieval in fact would have answered the question due to association; in any case, without the actual answer, at least a phone number came back!

In the fourth case, the results were also encouraging. Consider the query "Find the memo which lists the proposed locations of public ems in the usa." The system retrieved two documents with high relevance weights, both of which dealt with proposed public ems in the UK! There were no documents relating to the same in the USA. However, if there had been documents relating to both, the 'USA' documents would have been at the top of the list, with the UK running close second. By this one can see the tendency of the system to collect related groups, often a valuable feature to the mail system user.

In all cases, the demarcation between relevant and irrelevant documents was very distinct, with obvious cutoff points. One problem, in fact, was that the retrieval worked too well to gain any significant information from relevance feedback experiments.

An example of the retrieval process follows:

Consider the query -
WHAT IS HAPPENING IN THE DEVELOPMENT OF A HUMAN FACTORS COURSE?
This became (after stripping)

HAPPEN	DEVELOP	HUMAN	FACTOR	COURS
1/58	1/58	4/58	4/58	8/58
		2/61	2/61	4/61

The numbers below the query indicate the frequency of the term and the document number. It can be seen that term frequencies may

provide useful information for some queries in the business environment. However, the storage overhead involved in retaining these weights may be a more important factor. Document 58 had one or more occurrences of every key term in the query, and document number 61 had three term matches. As you might guess, document 58 addressed the development of a human factors course specifically, while document 61 (and actually a couple of others) talked about the need for, etc., a human factors course.

It is of note that these queries were processed without any access to header information, and accurately retrieved documents nonetheless. There did not appear to be any general need for a synonym matcher for these query collections.

2.11 Document clusters (R5)

The algorithm mentioned in section 2.10 was used on the H collection. It produced 15 clusters ranging in size from 2 to 8 documents. Of these 15 clusters, 9 were judged (by inspection) to contain groups of documents that a human might identify. Therefore this classification procedure may be of some use in determining useful groups of documents. This topic will be discussed again in section 3.

3. Recommendations for a text filing and retrieval system

In this section we shall summarize the results presented in the form of a proposed design for a text filing and retrieval system. The design will be described at a number of levels. These levels include the functions that the system will perform, how the system will be integrated with the other tools in the office information system, a discussion of possible implementations of the system functions and, finally, the type of workstation that would be suitable for the proposed system.

3.1 Functionality

The main functions of the text filing and retrieval system are as follows

File

Retrieve

Display

Delete/Archive

Classify

Dictionary

Lend

Of these functions, file and retrieve are the most complex. The objects that these functions deal with are documents. There are a number of document types the system will know about (such as memos, letters, elctronic mail, bibliographic references and forms), but all documents consist of a number of fixed fields and text fields. Fixed fields contain a single piece of information that is readily

identified such as a name, a date or an order number. They are called fixed fields because, for a given type of document, these fields will occur in fixed locations (such as the header information in a mail message). The text fields also occur at definite places in the documents, however the information they contain (namely, text) is much more difficult to identify and label than a fixed field. Some examples of types of documents are

Electronic mail or letters	Fixed fields	Source, Destination Date, CC
	Text fields	The message
Report	Fixed fields	Author, Date
	Text fields	Abstract, text
Bibliographic References	Fixed fields	Author, Date, Journal Pages, Volume, Location
	Text fields	Abstract
Order Form	Fixed fields	Crder Number, Supplier Part numbers, Quantities
	Text fields	Comments

The last type of document is typically handled by a database system since the most important information is in fixed fields. More will be said about the interaction with database systems later. The definition of a document given here could be extended slightly to include other types of information kept on the computer such as programs, data files and command sequences.

File

This function involves the storage of documents and any associated information. The filing of the fixed fields of a document is straightforward because, if a document is stored as a record in some file organization, then each fixed field will be a primary or

secondary key for that record. However, in order to answer a wide range of queries about document content, the text fields of a document will be indexed. That is, keywords representing the content of the text will be added to the document record. Indexing can be done automatically or by interaction with the users. All documents which are filed will contain keywords derived automatically from the text fields. They may also contain keywords which have been assigned by the user. This allows the user to set up their own categories of documents (electronic folders) and, because of the automatic indexing, the documents will still be filed in a consistent manner throughout the organization. The consistent filing policy will mean that the documents of different users and different departments will be accessable to any user on the system assuming security criteria are met. The specification of access rights for documents and any specific deletion/archive dates will be done through the filing function of the system.

The file function should establish where documents are to be filed. For example, some types of documents may be filed automatically at the departmental level (departmental memos) whereas other types may only be filed by individual users (casual messages, bibliographic references). Another important piece of information for a document that should be recorded by the file function is a list of any related documents previously stored. This will be mentioned again in the discussion of electronic mail.

Many documents in the office will not be in machine-readable form. For example, outside reports, journal papers and books. This type of document can be handled by filing bibliographic references to them. Indexing would be done by entering an abstract or keywords

for the document. Provision should also be made for indicating the physical location of the document in the office and for recording whether a document has been borrowed and by whom.

Finally, the file function should allow for the fact that the contents of documents are constantly modified and that multiple copies of documents will be stored in the system.

Retrieve

This function will provide a number of ways of specifying queries about documents and a means of integrating these types of queries.

Queries involving fixed fields are usually in the form of Boolean specifications. For example, "retrieve the documents written by Jane Smith in 1979-1980." In general, it should be possible to provide a query language with the power of, say, relational calculus to deal with these fields. This type of query includes questions about the contents of personal "folders".

Queries involving the content of documents as expressed by their text fields can be specified as follows

(a) Natural language

(b) Sets of keywords possibly including Boolean operators

(c) Examples of documents with similar content

An essential part of the retrieval for this type of query will be the relevance feedback process described earlier.

The system should provide the ability to specify integrated queries involving both fixed fields and text fields. For example, "retrieve documents written by Jane Smith about text manipulation." The fixed fields can be used to narrow the scope of the search either before or after searching by content.

A method of browsing the information stored in the database would be desirable - this would be a much less constrained search and it would only be used if the user has difficulty finding documents or expressing the query.

Display

This function will provide the user with a number of options for displaying fields of documents or sets of retrieved documents, either using graphics or printer output.

Delete/Archive

This function allows the user to delete documents from the database or to specify that certain documents are to be archived (stored on a long-term off-line medium). Provision will be made for specifying when deletion/archival will occur at the time of filing the document and different types of documents will have different default deletion/archive policies. The system will maintain an index to the archived documents so that they can be searched if necessary. When documents are deleted, they will be put into an electronic "wastepaper basket" which will be able to be searched until the contents are periodically removed.

Classify

The file function allows the user to attach keywords or "folder names" to documents when they are filed. The classify function will be used to define groups of documents which can be used for filtering incoming mail or for selective dissemination of information (SDI) by the company library. The characteristics of

the required groups will be obtained by interaction with the user. The process will be semi-automatic in that the user will be able to define groups using keywords or combinations of fixed fields and the system will be able to suggest possible groups or derive group characteristics from example groups.

Dictionary

This function wil permit the user to modify the system's vocabulary. These modifications will include adding words to the vocabulary, indicating that words are in special categories (such as stopwords or proper names) and indicating relationships between words, such as the fact that two words are synonyms or that a word is an abbreviation for another word. This function may be evoked by the system when it comes across new words.

Lend

The lend function will be used to indicate who has borrowed documents which are stored as bibliographic references in the system. That is, this function will be used to keep track of books, papers and reports stored in the office. The same function will permit inquiry as to the status of documents and the printing of lists of borrowed documents.

3.2 Integration with other tools

The text filing and retrieval system will be part of an office information system and as such it will make use of, and be used by,

other tools in the system. The tool which is closest in function to the text filing and retrieval system is the database system. The database system provides the ability to define, store and retrieve records which are essentially collections of fixed fields. Therefore this aspect of text filing and retrieval could be handled using a database management system. Actually, for reasons of efficiency and simplicity, all filing and retrieval in the office information system should be implemented using the same primitive operations. This could be done by implementing the entire text filing and retrieval system using a database management system [10]. However, these systems involve a large amount of overhead for a small system and it may prove more efficient to provide a set of primitive operations which can be used by both the text filing and retrieval system and a simple database system.

The editor of the system will be integrated with the text filing and retrieval system in that the indexing can be done in a cooperative manner. The editor will provide information to the indexing process and it will, in turn, interact with the user through the editor. Sophisticated editors which have knowledge about the structure of the documents [11] will be able to provide more information. Most of this information will be in the form of additional fixed fields.

The spelling corrector is used in text preparation. Typically, this tool uses a dictionary of words [12]. The text filing and retrieval system can use the same dictionary during the indexing process as a stopword list and a thesaurus.

The electronic mail system is the major method of communication using documents. When documents are sent using this system, the

user will provide the fixed fields that make up the mail header as well as some optional information such as where the document should be filed. If a user is replying to a message, this will be indicated to the filing system so that the documents may be linked. If a message is not wanted after being read, it will be deleted to the "wastepaper basket".

An important tool which has not received much attention is the procedure specification tool. This tool would be used to specify common procedures in the office [13]. These procedures would contain many tool instantiations including uses of the text filing and retrieval system.

3.3 Implementation of the system

The main impact of the experiments described earlier is on the design of specific indexing and retrieval strategies. The experiments showed that the simple indexing strategy mentioned in section 1.1 is appropriate for the office environment. Some small modifications can be made, but they will not have a major effect. For example, the majority of the two-letter words can be ignored, but the system should be able to keep a small list of important two-letter words. In most cases this list will be empty. The alternative is to put two letter words into the stoplist and keep all those not mentioned. This option will take more time during indexing and will produce longer document representatives. Peoples' names, although very common, can still be treated as normal keywords. Some tolerance to spelling errors in fixed fields such as

destination or author could be provided by the use of Soundex codes, although this should not be necessary.

Although all digits were removed from the text in the experiments, the system should have the capability of keeping strings of digits such as "1979" or "370". The decision as to whether to keep this digit strings could be left up to the user.

Extra stopwords can be specified using the dictionary function. The experiments indicate that this facility would be used infrequently. Abbreviations and synonyms will also be specified this way.

One of the most important results of the experiments was that documents of widely varying lengths can be represented by fairly small sets of index terms by deleting those terms with low weights. After these terms are deleted, the remaining term weights can be treated as binary or retained. Binary weights save storage and are almost as effective, so this option will be used.

The retrieval technique used will be inverse document frequency weighting combined with relevance feedback if necessary. This technique has been shown to be consistently better than the simple match used in the experiments. In some special cases it may be appropriate to be able to search the full text of the documents for specific strings. This facility could be provided by using one of the extremely fast string-matching algorithms recently developed.

Automatic document and term classification methods appear to be of limited use. Therefore the process of defining categories of documents for filtering mail and SDI will rely heavily on interaction with the user. Simple classification methods for

documents may be useful in determining some obvious groups automatically and the same applies to term groups.

3.4 The Workstation

The workstation on which the text filing and retrieval system is implemented would ideally have the following characteristics

1. A large amount of local processing power to run a range of tools.
2. Significant amounts of local storage (including Winchester disk).
3. High resolution graphics screen with windowing capability.
4. A variety of sophisticated software such as graphics-based text editors.

Apart from the high-resolution graphics and some of the software, most of the properties of such a workstation could be simulated by a terminal connected to a central computer facility. However, a major disadvantage of not having the graphics would be that the user interface would not be nearly as convenient or easy-to-use. In fact, some aspects of the system may depend on graphics for acceptance by users.

4. Conclusion

Statistical techniques designed for the filing and retrieval of scientific documents appear to perform well in the business environment with little or no modification. The effective use of this tool does, however, depend on its integration with other tools in an office information system.

References

1. Salton, G. "Automatic information organization and retrieval". McGraw-Hill, New York (1968).

2. Van Rijsbergen, C.J. "Information retrieval. Butterworths, London (1979).

3. Sparck Jones, K. "Automatic indexing". Journal of Documentation, 30: 393-432; 1974.

4. Sparck Jones, K. "Research on automatic indexing 1974-1976". British Library Research and Development Report 5464, Computer Laboratory, Cambridge (1977).

5. Landauer, C; Mah, C. "Message extraction through estimated relevance". Proceedings of the 2nd International ACM SIGIR Conference (1979).

6. Harper, D.J. "Relevance feedback in document retrieval systems: An evaluation of probabilistic strategies". Ph.D. Thesis, University of Cambridge (1980).

7. Croft, W.B. "Organizing and searching large files of document descriptions". Ph.D. Thesis, University of Cambridge (1979).

8. Croft, W.B.; Harper, D.J. "Using probabilistic strategies with no relevance information". Journal of Documentation, 35: 285-295; 1979.

9. Sparck Jones, K. "Research on relevance weighting 1976-1979". British Library Research and Development Report 5553, Computer Laboratory, University of Cambridge (1980).

10. Porter, M.F. "Implementing a probabilistic information retrieval system". Information Technology: Research and Development, (to appear).

11. Proceedings of the ACM SIGPLAN SIGOA Symposium on Text Manipulation, (various papers) June, 1981.

12. Peterson, J.L. "Computer Programs for detecting and correcting spelling errors." Com.ACM, 23: 676-687; 1980.

13. Ellis, C.A.; Nutt, G.J. "Office information systems and computer science." ACM Computing Surveys, 12: 27-60; 1980.

Appendix 1 : Two-letter word analysis.

Total number of two letter words : 266

most common	number of occurrences
cc	65
re	19
am/pm	13
do	12
mr/ms	7

Others :

Locations : NY,NH,CA,MA,LA,TX,MT,VA,UK,ML,MK,
TW,WM,SP,PK,AB

Obvious : GO,HI,ID,HR,VS,JR,DR,FM,ST,OK,TV,
PS,RT,EX,PC,TH

Names : ED,AL,BJ,KO

Numerical : II,IV,VI

Computer field : DP,WP,IO,LF,KB,SE

Unknown (used once) : AA,AQ,AY,BC,DA,ER,FT,GP,ND,
PE,UN,VE,VI,WK

Unknown (used more than once) : EM(2),ET(2),LC(3),
OP(2),SB(2)

Appendix 2 : Collection statistics.

Document Statistics

Collection	No.docs	Postings*	Av.length	Av.length after stopword deltn.
D	249	7285	29.3	26.3
H	75	5120	73.7	68.3
G1	85	4554	53.6	48.3
G2	43	7233	168.2	159.1

Length of doc.	Collection							
	D	D'	H	H'	G1	G1'	G2	G2'
0-2	3	6	1	1	1	1	0	0
2-10	36	45	7	8	15	22	0	0
10-30	143	146	15	21	34	29	2	2
30-50	44	34	18	14	12	13	1	1
50-70	9	3	10	9	6	7	3	3
70-90	5	6	6	8	6	2	6	9
>90	11	9	18	14	11	11	31	28
Max.length	263	228	551	528	556	512	513	508

* Postings is the sum of all document lengths

' Documents after extra stopword deletion

Appendix 2 (Continued).

Term Statistics

Collection	No.terms	Av.terms per doc.	No.terms after stpwd. deletion	Av.terms after deltn.
D	2247	3.2	2110	3.1
H	2085	2.7	1983	2.6
G1	1881	2.4	1753	2.3
G2	2254	3.2	2130	3.2

Length term lists	Collection							
	D	D'	H	H'	G1	G1'	G2	G2'
1	1249	1204	1102	1062	1019	963	1161	1104
2-5	695	647	752	713	680	635	735	688
5-10	170	141	148	135	136	117	216	202
10-20	88	80	73	66	45	37	115	110
>20	44	38	10	7	1	1	27	26

Appendix 3: Frequency counts within documents.

Frequency count	Collection			
	D	H	G1	G2
1	5801	3953	2848	4428
2	546	682	630	1316
3	115	227	240	542
4	32	119	121	292
5	18	49	70	140
5-10	22	65	122	329
>10	8	25	78	186

Appendix 4: Document statistics after term deletion

Document length	Collection					
	D	H	G1	G2	G2(2)	G2(3)
0-2	187	22	41	1	3	7
2-10	47	25	24	2	10	14
10-30	11	19	11	13	14	13
30-50	3	5	1	9	7	4
50-70	1	1	4	4	3	1
70-90	0	0	1	3	1	1
>90	0	3	3	11	5	3
Max.length	57	178	257	246	168	114
Av. length	3.0	15.6	14.8	62.7	34.6	22.0

All results are with terms of weight 1 deleted except for G2(2) and G2(3) which have terms of <=2 and <=3 deleted respectively.