Automatic retrieval and dissemination

Hoyle, W. G.

Publisher's version / Version de l'éditeur:
https://doi.org/10.4224/21276330
Report (National Research Council of Canada. Radio and Electrical Engineering Division : ERB), 1971-10

ERB - 859

UNCLASSIFIED

NATIONAL RESEARCH COUNCIL OF CANADA

RADIO AND ELECTRICAL ENGINEERING DIVISION

AUTOMATIC RETRIEVAL AND DISSEMINATION

W. G. HOYLE

OTTAWA

OCTOBER 1971

ABSTRACT

Those responsible for storing and indexing information appear best fitted to handle dissemination and retrieval. An earlier paper describing a computer procedure for the first two is here extended to describe a procedure for the latter two. Fully automatic (no editing or dictionary) assignment of keywords to documents is described and examples are given. The techniques are statistical but word counting is not used. The techniques are independent of meaning and therefore of language, but inflected languages such as Russian or German may alter results. A new measure of system performance is proposed based on the number of transpositions needed to bring the list of output items into agreement with the users opinion. Essential to this concept is the belief that any information system should present an ordered output and adjustable threshold. Relevance is reserved as a mathematical measure. An appendix describes some mathematical ideas useful in system development. Keywords selected by the computer from the above abstract: 08 dictionar, 075 editing, 075 languages, 07 indexing, 07 retrieval, 05 language, 03 mathemati.

CONTENTS

TABLES

AUTOMATIC RETRIEVAL AND DISSEMINATION

[Humanities]

—Ah les bon vieux temps où
nous étions si malheureux!—

– W.G. Hoyle –

Introduction

An information system should do at least two things, provide relevant material upon request and alert its users to incoming material relevant to their stated interests. Now it seems a reasonable assumption that those responsible for indexing and storing the incoming material are best fitted to perform these other functions. In an earlier paper [1] we showed that a computer could do classifying and indexing and therefore it seems reasonable that it should also do retrieval and dissemination.

First we consider the indexing procedure of reference 1 in somewhat more detail. We indicated that a computer could either take over an existing system or could establish its own categories along the lines suggested by Doyle [2], perhaps guided by our own ideas [3]. However Borko and Bernick [4] indicate that a computer does about equally well with a computer derived classification or a manually derived one, and since the latter is the more likely in practice, we have followed that plan in our experimental work. The computer generates a list of words, with attached probabilities, to represent each category. A sample list is shown in Table 1. This list represents the category 08, "Mathematics" in the scheme used by the IEEE for classifying abstracts on computers. It was generated from about 140 abstracts (10,000 words) representing categories 1 to 9 inclusive. Words less than four letters are not used (to save money) but other than this the list is prepared entirely without exercise of human intelligence. The probabilities attached to each word are computed from a theorem of Bayes, also without human intervention, and we thus avoid the serious error noted by Taube [5] "..... *the illegal shift from subjective relevance, as a reaction of a user, to mathematical relevance.....*". The use of word lists to represent categories is not new. Uhlman [6], refers to 'categoric descriptors' and in a later paper [7] describes the application of probability measures to the lists. Assorio [8] calls the lists 'average field vectors' while Salton [9] refers to them as 'centroid vectors'. Our lists differ in that both words and probabilities are derived entirely by the computer, and are thus uncolored by subjective judgment and are, hopefully, cheaper.

To index a document, its words are tried against the words of each category list in turn and the word probabilities summed within each category for each match. A sample result for 92 documents is shown in Table 2. (In the sample, classification is forced; i.e., rejection is not allowed).

The table tells us much more than the most probable category for the document (its primary function). It tells us also the probability of error, or how clearcut the decision is, and which documents or categories give rise to difficulties. It guides us in setting limits for

TABLE 1

CATEGORIC DESCRIPTORS (PART) FOR THE IEEE CATEGORY 08, "MATHEMATICS"

152	THREE
156	CERTAIN
156	DEVELOPED
159	ALSO
159	RESULTS
165	ERROR
165	MORE
165	NUMBER
165	THAN
165	VERY
172	SUCH
177	PAPER
180	DEFINED
180	ONLY
–	
–	
–	
–	
399	EXCEPT
399	NUMBERS
399	OBTAIN
399	RESULT
415	PROPERTIE
427	ERRORS
427	LEAST
427	METHOD
460	DIFFERENT
499	COMPUTE
499	DEGREE
499	DEPENDS
499	GENERALLY
499	NUMERICAL
499	STARTING
499	STEPS
499	STUDIED
544	APPROXIMA
599	INTEGRATI
599	PROPERTY
599	SOLVING
665	EVALUATE
665	EXPANSION
665	GENERATIN
665	ITERATIVE
665	OVERALL
665	POINTS
665	SQUARES
665	STEP
665	TAKES
665	UNIFORM
713	POLYNOMIA
749	COMPUTED
749	CONNECTIO
749	GENERALIZ
749	PARTIAL
749	PROVED
749	VALUES
799	EQUATION
999	CONCERNIN
999	DOES
999	FORMULA
999	INDEPENDE
999	INITIAL
999	INTERPOLA
999	POSSESS

10041	Total Words
7648	Total Tokens
838	Category Tokens
1195	Total Word Type
131	Category Word Types

rejection or cancellation and in general directs human abilities to where they are needed instead of employing them in routine indexing. The calculation of the table is described in the Appendix. Categories can be altered by manually forcing documents into a category where they would not normally belong. The computer will eventually alter the category lists and the altered subject area will be maintained. Categories will drift with time in any case. The performance, 54 of 92 documents correctly indexed, (i.e., consistent with the professional indexers of the IEEE) is good. Hooper's [10] review on consistency gives values of 10 to 80% as existing in the literature.

For retrieval, a query is treated first exactly as a document. It is indexed (retrieval and indexing are basically related) and directed to the most probable category. The total contents of the category are usually too large to offer as an answer and a further step is necessary to control the amount of material retrieved.

TABLE 2

CLASSIFICATION OF ABSTRACTS BY THE COMPUTER, WITH HUMAN DISAGREEMENT SHOWN

IEEE ABSTRACT	CATEGORY NUMBER								
	01	02	03	04	05	06	07	08	09
6763	0	25	32*	42	35	4	26	23	17
6764	38	21	116	43	19	16	19	12	7
6768	8	29	11	54	28	20	27	8	29
6769	17	37	26	40*	61	39	11	12	20
6782	32	4	11	27	61*	67	16	30	25
6783	39	0	22	17	73	27	21	28	29
6789	32	0	8	14	0	53*	40	18	54
6790	9	27	16	49	50	42*	46	8	29
6791	4	5	4	22	23	54*	55	9	5
6792	56	6	37	32	36*	13	13	38	12
6793	42	0	0	30	23	11	70	3	13
6797	36	6	12	20	10	16	49	88	17
6798	14	18	13	18	19	0	21	22*	29
6807	40	19	13	11	27	19	37	27	53
6808	32	40	18	11	17	0	6	23	5*
6809	6	9	18	29	27	0	21	22	38
6810	34	41	44	30	16	14	4	10	40*
6684	20	5	81	69	0	0	29	9	17
6685	10	24	130	54	33	8	20	0	13
6687	12	28	81	73*	21	6	33	16	20
6688	0	28	46	77	16	0	16	6	27
6689	6	22	19	58	56	0	18	9	72
6690	16	7	26	26	62	26	35	7	8
6691	10	24	30	15	70	18	11	18	15
6692	25	0	8	25	131	0	0	3	65
6706	27	4	12	19	22	72	19	4	0
6707	35	9	23	21	34	60	19	21	29
6708	18	5	20	20	38	96	35	6	17
6709	76	13	15	22	21	22	49*	20	28
6711	48	14	18	17	26	15	105	51	36
6716	45	7	0	35	27	25	27	39*	67
6717	14	9	13	23	46	0	30	65*	70
6718	13	8	43	40	39	0	12	67	57
7255	17	15	29	37	10	50	9	14	27

Values should be divided by 1000 to obtain actual probabilities.
Underlined values are maximums. Asterisks indicate disagreement.

TABLE 2 (Cont'd)

IEEE ABSTRACT	CATEGORY NUMBER								
	01	02	03	04	05	06	07	08	09
7256	25	6	17	11	43	70	14	16	10
7257	22	0	11	32	35	29*	25	3	28
7272	49	4	36	27	23	13	15	19	4*
7430	104	0	0	24	17	31	23	6	22
7431	141	16	0	11	0	63	17	39	5
7432	92	5	17	9	15	4	14	40	17
7433	106	0	9	9	10	5	37	49	21
7444	20	99	11	39	28	8	20	12	11
7445	15	105	49	29	15	3	30	27	8
7446	13	52	29	34	21	0	13	3	3
7447	27	109	33	5	14	36	18	5	4
7455	11	4	116	65*	11	0	5	13	0
7456	3	45	65	72	13	9	18	7	31
7457	0	29	60	7	12	30	7	15	0
7458	7	23	31	53	26	17	19	37	26
7459	19	0	0	35	18	31	21	20	22
7460	5	6	76	46*	13	11	25	20	26
7461	12	26	34	30*	18	2	26	38	26
7467	41	18	11	21	35*	50	32	34	19
7468	19	4	7	36	58	59	11	20	21
7470	24	39	19	45	59	8	24	13	25
7488	12	35	30	27	25	101	24	3	13
7489	22	16	29	15	26	19	70	8	11
7490	48	0	26	27	13	9	52	36	11
7491	13	8	18	36	24	15	34*	15	2
7492	75	12	30	11	17	56	33*	27	18
7495	35	24	7	6	28	0	24	77	48
7496	47	24	9	57	9	0	27	48*	25
7497	39	16	26	13	8	18	17	63	55
7498	29	0	0	19	17	22	0	20*	10
7503	13	4	56	20	44	7	35	4	47*
7509	33	11	19	17	27	23	22	19	19
7510	38*	0	16	39	44	16	26	34	9
7511	37	26	22	30	35	11	34	5	18

Values should be divided by 1000 to obtain actual probabilities.
Underlined values are maximums. Asterisks indicate disagreement.

TABLE 2 (Cont'd)

IEEE ABSTRACT	CATEGORY NUMBER								
	01	02	03	04	05	06	07	08	09
7512	20*	0	13	11	55	35	0	6	6
7518	12	72	38	46	30	16	0	0	23
7519	39	20*	44	37	40	8	12	30	28
7520	8	17*	34	51	25	21	8	34	3
7521	7	25*	25	49	44	9	14	10	5
7525	5	26	61*	92	15	4	55	5	22
7526	4	24	102	31	20	0	5	14	19
7527	21	24	53	33	27	0	7	26	15
7528	26	9	61	35	12	18	4	26	27
7531	19	19	19	32*	19	19	19	19	43
7532	15	25	43	84	11	12	14	8	11
7533	8	10	69	39*	18	5	12	19	15
7543	8	18	42	39	49	0	26	3	12
7544	7	28	30	45	54	8	17	30	4
7545	14	33	24	48	49	30	18	10	13
7546	50	0	24	19	39*	20	33	26	8
7575	6	3	30	38	55	40	38*	3	0
7576	38	5	10	28	19	14	42	0	22
7577	14	7	12	10	17	20	42	61	24
7578	18	0	0	32	8	19	18	103	38
7579	39	5	17	29	35	37	17	19*	10
7580	13	23	26	22	10	25	43	18	9*
7581	19	32	0	58	25	0	0	19	63
7582	21	9	12	23	69	29	29	9	31*

Values should be divided by 1000 to obtain actual probabilities.
Underlined values are maximums. Asterisks indicate disagreement.

For discrimination among the documents of the category, the computer (at the time of indexing) attaches keywords to the documents. The keywords are those words whose match with the category list resulted in the maximum probability sum and thus caused the document to be assigned to its particular category. Keyword weights are retained. Four sample lists are shown in Table 3. In retrieval after the query has been directed to the most probable category, it is passed in turn against each of the document word lists in the category and the probabilities summed for each document. The documents are then marshalled in order of the probabilities obtained, and presented. If too few references are retrieved the second most probable category can be processed and so on. It is of importance that the system, if unchecked, will present all of the collection material in order of probability. To limit the amount, a threshold must be set by the user. The threshold is usually in some form of minimum match requirement and its establishment is usually an adaptive process, preferably utilizing a visual display and on-line feedback.

TABLE 3

KEYWORD LISTS AS GENERATED BY COMPUTER FOR THE CATEGORY HAVING OPTIMUM MATCH

```
     26    21   41    29    26    86   32    40    26

ANALYSIS  ASPECTS   BASED     BEEN      CONCLUDED CORRESPON DICTIONAR EFFECTIVE
ENGLISH   ENTRIES   EXPERIMEN FROM      GRAMMARS  HAVE      LANGUAGE  MEANING
NATURAL   QUESTION  QUESTIONS RESEARCH

     73    33   17    34    33    34   94    40    31

APPROXIMA BASIC     CLASSIFIE FROM      INFORMATI INTRODUCE LEARNING  PATTERN
PROBLEMS  REQUIRED  RESEARCH  TECHNIQUE THEORETIC USING     VIEWPOINT

     51    28   16    37    84    56   50*   38    48

ALGORITHM DEBUGGING EFFICIENT INTO      LIMITED   NUMBER    OPERATING PROGRAM
PROGRAMME PROGRAMS  SEVERAL   SHOWN     SYNTAX    SYSTEMS   TABLES    TECHNIQUE
THESE     TIME      WHICH

     42    45   20    52    44    30   46*   41    30

BEING     COMMUNICA COMPUTER  DISTRIBUT EACH      LATTER    MAKE      MODEL
PARTICULA PROCESSOR RESULTS   SERVICE   SIMILAR   SIMULTANE SYSTEM    SYSTEMS
THIS      TIME      TIME-SHAR WHICH

     39    17   23    20    16    10   26    133   37

COEFFICIE COMPLEX   COMPUTED  EQUATIONS ERROR     FINDING   MATRIX    METHOD
POLYNOMIA REAL      ROOT      ROOTS     TESTS     WITH

     46    20   36    30    40     5   46    120   43

ALGORITHM ALWAYS    APPROACH  CALCULATI CONSIDERA CONSIDERE CONVERGEN EFFICIENT
EQUATIONS GIVEN     METHOD    METHODS   PAPER     POLYNOMIA PROPERTIE RATE
ROOTS     SOLUTION  SOLVING   THAT

     68    30   22    25    45    32   34    72    38

ALGEBRAIC ALGORITHM CONTAINS  EQUATIONS LINEAR    PAPER     RESULTS   SOLVING
SOME      THERE     WITH

     38    47   49    46    40    24   42    49    54

ANALYSIS  APPLICATI COMPUTER  DEVELOPED DIGITAL   INTERACTI INTRODUCE LINEAR
PROBLEM   PROBLEMS  PROGRAM   RANDOM    SHOW      SIMULATIO STATISTIC TECHNIQUE
THAT      TIME      TRANSMISS USED
```

Individual weights for keywords are not printed because of space limits. Values shown should be divided by 1000 to obtain actual probabilities. These values represent sum weights for the category divided by the number of words in the document or category, whichever is less. Underlined values are the computer choice (maximum). Asterisks show where the IEEE professionals disagreed. Abstracts are 7781, 7783, 7765, 7767, 7786, 7788, 7799.

Every librarian will recognize that the computer procedure is exactly that followed in their manual procedures. The computer need not examine every document in the collection – a serious defect in many mechanized systems. To quote Goffman [11 p. 361], "Two important defects–one affects the efficiency of the system in that the entire set of documents.... must always be probed....".

Selective dissemination (SDI) requires a somewhat different procedure. Since the documents to be searched have not yet been entered into the system (SDI is done on incoming material) the procedure is reversed and the user profile is indexed. A word list is thus generated for each category and the most probable selected. (The profile does not enter into the preparation of category lists. The system can always put out a list of the matches of an item against any or all categories–such a list can be very useful for studying the system.) This optimum word list for the profile is matched against the incoming material which is then marshalled by probability and presented up to a stated limit. If the quantity is insufficient, the second most probable category is processed and so on. We have no sample of this SDI procedure. It is our most recent development and a very tentative process. We consider it the most important aspect–it could be used, for example, to guide the library on acquisition, given the library profile.

In conclusion, there are several things to be said. We do not employ word counting; in fact, our first step is to eliminate all word repetitions in a document. The program, however, can demand instead of one, a minimum of two (or any number of) occurrences in a document. Using whole text, rather than abstracts, some number other than one may be preferred; perhaps it should be based on the length of document. We do not verify our keypunching; therefore optical character recognition devices, which are almost essential for an operational system, and which will have presumably a substantial error rate, should not cause us any trouble. Abstracts have been used instead of full documents merely to save money and to increase maneuverability in an experimental situation. The procedures are independent of word meaning and, therefore, of language, but highly inflected languages such as Russian and German may not give identical results. Indicatively, there is apparently greater difficulty in preparing KWIC type indexes in German [12, p.40]. Russian, with its additional wealth of prefixes, challenges a comparison.

The small amount of material in our trials makes some of our conclusions of dubious value. A full operating system would be ideal, if one were permitted to experiment with it.

We have formed two general ideas from our experience with automatic systems. First, relevance, as we use it, is merely a mathematical measure which we use in marshalling documents. We propose a new and different measure by which users should judge a system. All documents are relevant in some sense, and an ideal system merely presents documents in the order preferred by the user. There must be also a means of setting a cut-off as generally something less than all documents are wanted (depending on the time available for reading before more material arrives). As a numerical measure of performance, the number of transpositions needed to bring the list into agreement with the user's judgment would seem suitable. In a more practical vein, since we usually cannot afford to examine the whole collection, we could examine a number of documents below the threshold equal to the number above and count the number of documents to be exchanged.

Our measure has the great advantage that it requires merely the comparison of two documents. Absolute relevance is avoided. The user agrees not only with what he is getting but to some extent on what he is not getting. Implicit in our measure is the inverse relationship of relevance and recall. The use of only the first part of the list in judging the system is supported by the work of Lesk and Salton, in particular [13, p.355], they say:"The conclusion is then obvious that although there may be a considerable difference in the document sets termed relevant by different judges, *there is in fact a considerable amount of agreement for those documents which appear most similar to the queries and which are retrieved early in the search process* (italics in original)..... Since it is precisely these documents which largely determine retrieval performance, it is not surprising to find that the valuation output is substantially invariant for the different sets of relevance judgments."

They go on to define the best possible delegated system as one in which ...."a subject expert completely reads through an entire document collection and *ranks each document in decreasing similarity order with a given search query*" (italics added). It is a moot point whether the questioner should be allowed to modify his requests in the light of previous answers before judgment is passed on the system performance.

Our second conclusion is that keyword selection is not the simple matter it seems. The classical view of indexing consists, at least in large part, in deciding for all time what part of the substantive content may someday be sought. Our program does not follow this precept, instead it decides what words distinguish a set of documents (maybe just one) from a larger set from which it is to be selected. In manual indexing this larger set must exist in the mind of the indexer, and some knowledge of the collection (e.g., whether it is a public library or a set of hospital records) is essential. Therefore an author or scientist may not be the proper person to select keywords if he is ignorant of the collection. A set of keywords for one collection may be totally unsuitable for another. The word *computers* may be an excellent keyword in a high school library, but totally useless in our context. This basis for selecting keywords is interesting in the light of present-day information theory.

We think our word lists may have some use in improving KWIC type indexing, if only in the preparation of 'stop' lists. Also, using the sentences in which document keywords first occur, we have prepared presentable abstracts of documents. We could display such an abstract to the user if he is unable to judge from the title if he wants a document, assuming, of course, no normal abstract is available and the complete document is too long.

The selection of vocabularies for specialized purposes appears to be another promising area of application. The selection of such vocabularies for professional groups or in foreign studies or for specific age groups in the native language (school grades) is not a trivial problem. See Alford [14]. As a simple example, here are lists which represent *girls* and *boys* as determined by the computer from a sample of four high-school essays. The vocabularies are not surprising; the relative lengths of the lists may be.

CAREERS
MARKS
STANDING

ARTS
CHILDREN
FAMILY
HUSBAND
LIFE
MARRIED
TEACHING

References

1. Hoyle, W.G. Automatic classification and indexing. Mezhdunarodny Forum po Infomatike, (International Congress on Scientific Information), Moscow, 1969.
2. Doyle, L.D. Breaking the cost barrier in automatic classification. Systems Development Corporation, Santa Monica, California, Special Publication 2516, July 1966.
3. Hoyle, W.G. On the number of categories for classification. Information Storage and Retrieval, 5: 1–6; 1969.
4. Borko, H. and Bernick, Myrna. Automatic document classification. ACM Journal, 10: 151; 1963.
5. Taube, M. A note on the Pseudo-Mathematics of relevance. Am. Doc., 16(2): 69; April 1965.
6. Uhlman, W. A thesaurus, "Nuclear science and technology": principles of design. TVF Teknisk-Vetenskaplig Forskning, Sweden, No. 2, 1967.
7. Uhlman, W. Document specification and search strategy using basic intersections and the probability measure of sets. Am. Doc., 19: 240–246; July, 1968.
8. Assorio, P.G. Classification space analysis. AD608034, RADC TDR 64 287 (AF 30(602) 3342). Rome Air Development Center, Rome, N.Y., October 1964.
9. Salton, G. Search strategy and the optimization of retrieval effectiveness in mechanized information storage, retrieval and dissemination. Proc. FID-IFIP Conf., p. 75, Rome, N.Y., June 14–17, 1967.
10. Hooper, R.S. Indexer consistency tests–origin, measurements, results and utilization. IBM Corporation, Bethesda, Md., 1965.
11. Goffman, W. An indirect method of information retrieval. Information Storage and Retrieval, 4: 361–373; 1968.
12. Stevens, Mary E. Non-numeric data processing in Europe, a field trip report, Aug.–Oct. 1966. U.S. Dept. of Commerce, National Bureau of Standards, Technical Note 462.
13. Lesk, M.E. and Salton, G. Relevance assessments and retrieval system evaluation. Information Storage and Retrieval, 4: 343–359; 1968.
14. Alford, M.H.T. Computer assistance in learning to read a foreign language. Information Scientist, 3(2): 63–68; July 1969.

APPENDIX

Our procedure for word matching is basically set intersection. The set of category words (as in Table 1) is the set P, the document or query words the set Q. The intersection

$$R = P \cap Q \tag{1}$$

we call the relevance or result. The expressions denoting general numbers of the sets are r_i, p_i and q_i. to each set member there corresponds a probability measure ℓ_i, m_i, n_i and

$$\ell_i = m_i \,.\, n_i \quad \text{if} \quad p_i = q_i, \quad \text{otherwise} \quad = 0$$

$$r_i = p_i = q_i \quad \text{if} \quad p_i = q_i, \quad \text{otherwise } r_i \text{ is null}$$

$$R = \frac{1}{K} \sum_{i=1}^{i=j} \ell_i \tag{2}$$

where K is the number of terms in p_i or q_i, whichever is less, j is the number of word pair matches between p and q

A similar procedure, though not identical, is described in reference 7. Table 2 was calculated using eq.(1) with $n_i = 1$. We also tried $n_i = m_i$ (i.e., taking the document word probabilities from the category list) but in two trials performance dropped from 14/17 to 13/17 and from 13/16 to 12/16 correctly indexed documents—not very significant but not encouraging. For a request the user can set his own document word weights (if he presents a list of descriptors rather than literary English as a request). The procedure in (1) can be reversed, so to speak, by taking the exclusive disjunction of P and Q rather than the intersection. The result Fairthorne [A] calls the 'distance' between P and Q. It represents the set whose members are members of either P or Q but not both (logical exclusive or). We may think of our R in (1) as a metric, but though it satisfies the symmetry and triangle inequality requirements it is not necessarily true for $R = 0$ that $P = Q$ and it is not a true metric, while that of Fairthorne would appear to be so.

Another approach to the matching problem is that of Wilde [B] who introduces the idea of a 'threshold gate' as an addition to Boolean logic to obtain Threshold Logic, a branch of Switching Theory. He adds that Threshold Logic is sometimes referred to as Weighted Term Logic in Information Retrieval.

Taulbee's approach [C] is through the equivalence relation, establishing a binary relation on a set. The equivalence relation separates the set into classes. Sammon [D] uses as a mathematical model a two-dimensional matrix, one dimension being identifiers, the other documents. The query is a one-dimensional vector. In brief we see that operation of word matching in (1) can be done in many other ways than we do in (2).

If we estimate the vocabulary for a given collection (at any one time, otherwise it is infinite) let us say 10,000 words, and if we assume a typical category list contains 120 words and the document vocabulary 80, then the probability that a word occurs in a document, given that it has occurred in the category, is $\frac{80 \times 120}{10{,}000} \cong 1$. We may therefore expect a 1 word match between category and document as a purely chance occurence. Thus we have a sort of 'noise level' as a guide to setting thresholds and making decisions.

When the words for category lists of the type in Table 1 reach about 10,000 (100 + documents) computer limitations enter. At this point a second set of lists, based on the next 100 or so documents is prepared and merged with the first set. New unique words are added with their full probabilities. Duplicates are removed, but the weight of the existing word is averaged with that of the removed word after all the existing weights are reduced by a decrement factor. Thus words which are not 'refreshed' gradually drop in value, as suggested by Dennis [E p.66] "....follow the progress of each word. As documents are added to the file......drop out a word when its criterian has passed.....a threshold". She also [p.65] does computations at intervals of 100 documents and suggests that 600 or 700 documents are sufficient to characterize the information, a useful basis for deciding our decrement factor.

In actual use, when doing comparison as in Table 2, documents are not forced into a category but have a rejection threshold (perhaps more than one). Mathematically this fact is important—we are forced from the field of Boolean into lattice algebra. Mainly the Boolean complement 'not' does not exist but rather there are two complements related to 'all but not only' and 'only but not all' in answer to a request. The difference is the rejects. See Hillman [F] for more information on these 'Brouwerian' and pseudo complements. Another useful reference is Litovsky [G].

References

A. Fairthorne, R.A. Delegation of classification. Am. Doc., 9(3): 59; 1953.

B. Wilde, D.U. Computer-aided stategy design using adaptive and associative techniques. Proc. Am. Soc. Information Science, 5: 175; Oct. 1968.

C. Taulbee, O.E. New mathematics for a new problem. *In* Electronic Information Handling. *Edited by* A. Kent. 1965.

D. Sammon, J.W. Some mathematics of information storage and retrieval. Technical Rept. No. RADC-TR-68-178, Rome Air Development Center, AD673362. June 1968.

E. Dennis, Sally F. The construction of a thesaurus automatically from a sample of text. *In* Statistical Association Methods for Mechanized Documentation, National Bureau of Standards Publication 269, p. 61, Dec. 1965.

F. Hillman, D.J. Mathematical classification techniques for non-static document collections with particular reference to the problem of relevance. *In* Classification Research , Elsinore, Denmark, Sept. 1964.

G. Litovsky, B. Utility of automatic classification systems for information storage and retrieval. University of Pennsylvania, Philadelphia, AD687140. May 1969.