polycyclic systems quite readily; other cases are passed to the more general but slower program described here. Both versions of the program are available for demonstration and experimentation over a nationwide computer network for those who wish to evaluate the program for their potential use.

Commented listings of the unmodified program (to which the above timing and core-requirement information pertains) are available from the author upon request. Special arrangements can be made for interested parties wishing to obtain a copy of the program in a more computer-accessible form. A version of the program adapted for the IBM 360/67 computer is also available.[11]

REFERENCES AND NOTES

(1) This work was supported by the National Institutes of Health, Grants RR00612-05A1 and RR00758-01A1; the latter in support of the Stanford University Medical Experimental Computer Facility, SUMEX.
(2) For an overview of recent work in these areas, see "Computer Representation and Manipulation of Chemical Information", W. T. Wipke, S. R. Heller, R. J. Feldmann, and E. Hyde, Ed., Wiley, New York, N.Y., 1974.
(3) See, e.g., Abstracts of Papers (COMP Division, Session on Computer Generated Graphics), 169th National Meeting of the American Chemical Society, Philadelphia, Pa., April 6–11, 1975, Port City Press, Baltimore, Md., 1975.
(4) B. L. Zimmerman, "Computer-Generated Chemical Structural Formulas with Standard Ring Orientations", Ph.D. Dissertation, University of Pennsylvania, Philadelphia, Pa., 1971.
(5) R. J. Feldmann, ref 2, pp 55–60.
(6) W. T. Wipke, ref 2, p 153.
(7) R. E. Carhart, D. H. Smith, H. Brown, and C. Djerassi, "Applications of Artificial Intelligence for Chemical Inference. XVII. An Approach to Computer-Assisted Elucidation of Molecular Structure", *J. Am. Chem. Soc.*, **97**, 5755 (1975).
(8) Although the typical spacing of characters in terminal printout is rectangular, we shall treat the grid as being composed of square elements to simplify the discussion. In the actual typed drawings, the "ideal" diagonal directions 45, 135, 225, and 315° are frequently closer to 60, 120, 240, and 300°, respectively. The atom states shown in Table I were composed using a computer terminal, and thus use the latter angles.
(9) Here, as throughout this paper, angles are measured counterclockwise from the positive x axis (the horizontal axis in the drawing plane).
(10) The vertical-bar symbol is available on many terminals and is preferable to the exclamation point. The typed drawings in this paper use the vertical bar.
(11) The "backslash" is peculiar to the ASCII character code, and there seems to be no suitable EBCDIC equivalent. The percent symbol (%) has been chosen arbitrarily for the version of the program adapted for IBM equipment (see section XI).
(12) With atoms of sufficiently high valence, CONGEN can generate structures with bond orders greater than three. The symbols * (asterisk), & (ampersand), and $ (dollar sign) are used for quadruple, quintuple, and hextuple bonds, respectively.
(13) D. G. Bobrow, J. D. Burchfiel, and R. S. Tomlinson, "TENEX, a Paged Timesharing System for the PDP-10", *Commun. ACM*, **15**, 135 (1972).

Selection of Descriptors According to Discrimination and Redundancy. Application to Chemical Structure Searching

LOUIS HODES

Department of Health, Education, and Welfare, National Institutes of Health, National Cancer Institute, Bethesda, Maryland 20014

Received November 14, 1975

Descriptors, for our purposes, will be fragment screens in a chemical search system. Given a file of compounds, the discrimination of a set of descriptors can be defined in terms of their incidence and mutual incidence in the file. A theory is developed which provides both a heuristic for selecting descriptors and a method for evaluating their marginal discrimination. These ideas have been used to generate an efficient screen code for a large file of chemical structures.[1]

1. INTRODUCTION AND BACKGROUND

Descriptors can be thought of as tags or labels applied to objects. They can be used to classify or retrieve subsets of the objects. It is in the nature of descriptors that they are usually used in looking for objects which contain them rather than for objects from which they are absent. In this way descriptors are philosophically different from ordinary binary variables where presence and absence generally carry equal weight.

We develop a theory of discrimination based on the incidence and joint incidence of descriptors, encompassing this nonsymmetrical property of descriptors. This theory is used to evaluate descriptors according to their marginal discrimination capability. This work is especially relevant to chemical structure searching, but it applies to taxonomy and information retrieval, i.e., wherever some selection of descriptors must take place.

This work is also related to feature selection in the area of pattern recognition.[2] However, features tend to occur in the form of variables rather than descriptors, this being sometimes also true in taxonomy work.[3] Descriptors in our sense of the term appear more frequently in document retrieval, but in that field the main problems are those of language, e.g., thesauri. Nevertheless, there have been some attempts to quantify the value of descriptors.[4,5] Closer in spirit to our work is that of Lee[6] and Kryspin and Norwich[7] on the use of information theory to select relevant variables.

Molecular structures supply a superabundance of good descriptors in the form of structure fragments,[8] and it is not easy to produce an appropriate subset. We develop here the information-theoretic concepts of discrimination, redundancy, and marginal discrimination of a new descriptor to supply the rationale for the construction of an efficient effective set of fragment descriptors used for screening a large file of molecular structures.[1]

Computers have facilitated the accumulation of large files of chemical structures, of the order of the hundreds of thousands or several million.[9,10] Because of these large numbers, searching the files according to structural characteristics has become a challenge.

It takes too much time to examine each structure in a file for a required substructure. The general problem is that of matching a graph to an arbitrary subgraph of another graph. It is almost certainly of exponential complexity. There are some good heuristics which take advantage of the nature of most chemical structure graphs, but even so, files quickly outgrew the size for which direct searching would be feasible.

Over years of experience, strategies for substructure searching of chemical files have evolved, some predating the use of computers. Many of these strategies are based on the use of fragments as descriptors, so that the chemical structures

as well as the substructure queries are coded in terms of the fragments they possess. These fragments are then used as criteria for quickly accepting or rejecting compounds for more extensive atom-by-atom searching.

The early stages of this evolution saw the proliferation of systems called fragmentation codes, based on sets of fragment descriptors. Each was custom designed for its specific application, and it has been estimated that the number of different fragmentation codes reached the hundreds.[11] Generally, as files grew in size and variety, most of these systems became obsolete and more complex codes had to be designed.

As the number of compounds in files grew into the tens of thousands, the principle of rapid screening was adopted. Under this principle, files are specially designed to facilitate substructure searches. This can be done in two ways. The fragments can be coded as a bit string so that the computer can rapidly reject compounds which do not have the fragments belonging to the query substructure. Alternatively, the fragments can form an index, and these "inverted" lists of structures can be intersected, one list for each of the query substructure fragments.

Next in the evolution of screening, attention was focused on the fragment screens themselves. It was early realized that, although there was a wide choice of fragments, their distribution among the compounds was far from ideal. A small number of fragments appears very frequently, in somewhat more than half the file. There are relatively few of moderate incidence, while the large majority appear in only a handful of compounds or just a single compound.

There was difficulty in establishing criteria to use in choosing fragments as descriptors.[12] Under conditions of independence (this will be defined in the next section) if a query has ten fragments each of which occurs in half of the file, screenout would be 99.9%, which is quite good, even for a fairly large file. However, it is not easy to find many fragments of about 50% incidence, and it was recognized that independence assumptions are unrealistic. Moreover, it is not clear that incidence as high as 50% is desirable. It is clear that if an extremely low incidence fragment occurs in a query, it is quite valuable for high screenout. On the other hand, any given low incidence fragment is not likely to occur in a query as there has been demonstrated to be a fairly good correlation between fragments occurring in queries and the file.[13]

The possibility of automatic generation of fragment screens from structures in the file was developed next. This emphasized the need for simple criteria in choosing a good screen set. Such criteria can be based on structural restrictions and can also be based on incidence and mutual incidence of the fragments in the file.

Purely structural criteria can lead to inefficiencies. For example, let us take X to be a rare element or a small fragment which occurs very infrequently in the file. Then X itself will have good discrimination, but any larger fragment containing X would be redundant when used along with X. Any algorithm which exhaustively generates all fragments of certain sizes cannot help running into this difficulty. Incidence information must be taken into account.

In an attempt to get an efficient screening system, the Sheffield group[14] has made a systematic, but limited, study of the incidence of fragments for their selection as screens. The bias of this group has been toward small fragments of high incidence. They bring to the forefront the important concepts of fragment incidence and mutual incidence. They also suggest the principle of forming new fragments by extending those of highest incidence. However, this principle was not systematized, the theory remained undeveloped, and their results proved unextraordinary. There is no doubt that if the Sheffield group had a really large file to search they would have gone further in the direction to be presented later.

2. THEORETICAL DEVELOPMENT

Suppose we have as our file a set of n objects to be searched, and a set of k descriptors. Without loss of generality we can use the numbers 1, 2, ..., k to represent the k descriptors. We define an item to be any subset of descriptors. Each object is represented by an item in the obvious way, and we will be considering only the collection of items corresponding to the objects. More than one object may be represented by the same item; i.e., duplicate items are allowable.

We let $K = \{1, ..., k\}$, the set of descriptors. Let A be any subset of K. The cardinality of a set A is denoted by $|A|$. Thus $|K| = k$. The set of items A is called $\Pi(A)$. Another useful designation is the set of items A' such that A' includes A as a subset. We call this set $\Psi(A)$. It is $\Psi(A)$ which is retrieved when the set of descriptors A is presented.

Also we will need the probability $P(A)$ of choosing item A when selecting an item at random. $P(A) = |\Pi(A)|/n$, the multiplicity of item A divided by the number of items. Most important will be $Q(A) = |\Psi(A)|/n$, the incidence of A, or the proportion of items in the collection which contain descriptors A.

We now draw from information theory the quantity $\log_2 n$ which we call the indefiniteness of our set of n items. This measure is used because it takes $\log_2 n$ bits to specify one out of any set of n objects. We shall define discrimination in terms of a reduction in indefiniteness. All logarithms will be assumed as base 2.

The set of discriptors A is normally used to reduce the number of items under consideration from n to $|\Psi(A)| = nQ(A)$. The remaining indefiniteness is thus $\log nQ(A)$ or $\log n + \log Q(A)$, which is a reduction of $-\log Q(A)$. Hence we can say that the discrimination of A is $-\log Q(A)$.

Now we can quantify the expected discrimination, which we will also call the discrimination of the entire set of descriptors. It can be expressed as the sum of the discrimination of each subset, weighted by its probability of occurrence. We assume here that the probabilities are the same as occur in the collection of items.

$$D(K) = \Sigma -P(A) \log Q(A), \text{ all subsets } A \text{ of } K \quad (1)$$

In the following we will simplify notation by eliminating brackets and commas from set notation. Thus $Q(\{1,2\})$ will be written $Q(12)$.

The concept of independence will be important. We say that two descriptors i, j are independent if $Q(ij) = Q(i)Q(j)$. More generally, the descriptors of a set A are mutually independent if any two disjoint subsets B, C of A, we have $Q(B \cup C) = Q(B)Q(C)$.

We say that two descriptors i, j are disjoint if $Q(ij) = 0$ and a set of descriptors A is disjoint if $Q(A) = 0$.

Example 1. Take the trivial case where there are no descriptors; $K = \emptyset$. Then, of course $P(\emptyset) = 1$, $Q(\emptyset) = 1$ and $D(\emptyset) = 1 \log 1 = 0$.

Example 2. Now suppose there is only one descriptor; $K = \{1\}$. Again $Q(\emptyset) = 1$ which always holds and will always nullify the contribution of the empty set to discrimination. In this case $P(1) = Q(1)$ and we get $D(1) = -P(1) \log Q(1) = -Q(1) \log Q(1)$.

From this formula for $D(1)$ we find that the maximum discrimination of a single descriptor is achieved when its incidence, Q, is $1/e$.

Example 3. Now we can consider the less trivial case, $K = \{1,2\}$. Going back to definition 1, we have

$$D(12) = -P(1) \log Q(1) - P(2) \log Q(2) - P(12) \log Q(12) \quad (2)$$

Note that

$$P(1) = Q(1) - Q(12)$$
$$P(2) = Q(2) - Q(12)$$
$$P(12) = Q(12)$$

hence we can rewrite (2) as

$$D(12) = -Q(1) \log Q(1) - Q(2) \log Q(2) - Q(12) \log Q(12)/Q(1)Q(2) \quad (3)$$

Note that the expression $Q(12)/Q(1)Q(2)$ plays a crucial role in eq 3. We denote this expression by $R(12)$. Recall that if descriptors 1 and 2 are independent, then $R(12) = 1$. If descriptors 1 and 2 are disjoint, then $Q(12) = R(12) = 0$. In both cases, the contribution to discrimination from the set {1,2}, which is the third term on the right-hand side of (3), vanishes. Thus, under either condition, independence or disjointness, the resulting discrimination is numerically equal to the sum of the descriminations which are obtained by means of descriptor 1 alone and descriptor 2 alone.

However, if $R(12)$ is in the range between 0 and 1, there is a positive, though small, contribution to discrimination from the set {1,2}. Thus, contrary to popular notion, independence is not always the ideal situation. In this case, disjointness is just as good, and the optimum value for $R(12)$ lies somewhere in between. For fixed $Q(1)$ and $Q(2)$ the optimum value for $R(12)$ is $1/e$.

Now $R(12)$ can also grow somewhat bigger than 1. Its maximum value is the smaller of $1/Q(1)$ and $1/Q(2)$. When $R(12)$ is greater than 1, it results in a negative contribution to discrimination. At maximum redundancy, $Q(1) = Q(2) = Q(12)$, the two descriptors are effectively identical and the contribution of {1,2} to discrimination, $-Q(12) \log 1/Q(1)$ equals $Q(1) \log Q(1)$, which is negative and cancels out one of the positive terms. This checks with our intuitive idea that two effectively identical descriptors have the same discrimination as one of them alone. The second descriptor is completely redundant.

Thus, we can think of $R(12)$ as a redundancy indicator for {1,2} based on {1} and {2}, with the following qualification. Values of $R(12)$ between 0 and 1 are good, but as we increase to values greater than 1 we get increasing redundancy.

Example 4. Let us now take a set of three descriptors, $K = \{1,2,3\}$. By definition 1

$$D(123) = -P(1) \log Q(1) - P(2) \log Q(2) - P(3) \log Q(3) - P(12) \log Q(12) - P(13) \log Q(13) - P(23) \log Q(23) - P(123) \log Q(123) \quad (4)$$

But

$$P(1) = Q(1) - Q(12) - Q(13) + Q(123)$$
$$P(2) = Q(2) - Q(12) - Q(23) + Q(123)$$
$$P(3) = Q(3) - Q(23) - Q(13) + Q(123)$$
$$P(12) = Q(12) - Q(123)$$
$$P(13) = Q(13) - Q(123)$$
$$P(23) = Q(23) - Q(123)$$
$$P(123) = Q(123)$$

Substituting into eq 4 we get

$$D(123) = -Q(1) \log Q(1) - Q(2) \log Q(2) - Q(3) \log Q(3) - Q(12) \log R(12) - Q(13) \log R(13) - Q(23) \log R(23) - Q(123) \log Q(123)Q(1)Q(2)Q(3)/Q(12)Q(13)Q(23) \quad (5)$$

Notice that the $R(ij)$ terms in eq 5 play a role similar to that of $Q(12)/Q(1)Q(2)$ in eq 3. The new term at the end of eq 5, $Q(123)Q(1)Q(2)Q(3)/Q(12)Q(13)Q(23)$, which we shall call t, can be analyzed more easily if one substitutes the $R(ij)$ for the $Q(ij)/Q(i)Q(j)$, getting $t = Q(123)/R(12)R(13)R(23)Q(1)Q(2)Q(3)$. We see that t reduces to 1 if descriptors 1,2,3 are mutually independent, since then all the $R(ij) = 1$ and $Q(123) = Q(1)Q(2)Q(3)$. Also, if {1,2,3} are disjoint, then $t = 0$. Thus, t has a similar effect, at the three descriptor level, to that of $R(ij)$ at the two descriptor level. The occurrences of $R(ij)$ in t seem to discount the effects of pairs of the three descriptors.

The foregoing regularities in the expression for $D(K)$ in examples 3 and 4 lead us to speculate that there may be a form of the expression for discrimination which offers some insight into the effects of independence and redundancy on discrimination. We shall show that there is precisely such a general expression, and then comment some more on its interpretation.

Given a set $K = \{1, ..., k\}$, with $Q(A)$ defined as before for all subsets A of K, we define $R(A)$ as follows. This generalizes the $R(ij)$ of examples 3 and 4.

Definition:

1. $R(\emptyset) = 1$
2. $R(A) = Q(A)/\pi R(B)$ all proper subsets B of A

Theorem: $D(K)$ of eq 1 can be expressed as follows.

$$D(K) = -\Sigma Q(A) \log R(A) \quad \text{all subsets } A \text{ of } K \quad (6)$$

The theorem will be proven in the Appendix. We will first use the inclusion–exclusion principle to express $P(A)$ in terms of $Q(A)$ and then use induction on the cardinality of sets A.

3. INTERPRETATION

As in the examples, eq 6 gives a quick interpretation of the relation of incidence and relative incidence to discrimination. $R(A)$ is again a redundancy indicator for the set A, and if it falls between 0 and 1 the set of descriptors A has a mutually positive effect on discrimination. We can think of a set A with $R(A) = 1$ as a pseudo-independence condition for that combination of descriptors. When $R(A)$ is much greater than 1, A has some redundancy among its descriptors and a better choice may be available.

We can interpret discrimination as in the examples by regarding $-\sum Q(A) \log R(A)$ for singleton A as the first-order discrimination, and this is the same as $-\sum Q(A) \log Q(A)$. The contribution for sets A containing more than one descriptor will of course be negligible if the descriptors are fairly independent and $R(A)$ is close to 1. Also, as A goes to disjointness, $Q(A)$ quickly gets quite small and $Q(A) \log R(A)$ again can be neglected. Thus, if descriptors that are not too reduntant can be avoided, then the first-order discrimination is a good measure of the total descrimination.

Equation 6 offers an advantage over definition 1 if we wish to evaluate the marginal discrimination of a new descriptor, x. The marginal discrimination is defined as the difference between $D(K \cup \{x\})$ and $D(K)$. This is easily seen to be the sum of $-Q(A) \log R(A)$ for all A such that $x \in A$. If we used definition 1, extensive recomputation would be required since $P(A)$ would generally change for all A. That is, $Q(A)$ and $R(A)$ do not change for $x \notin A$ whereas $P(A)$ will generally change when x is added. Even $P(\emptyset)$ will be reduced. In many cases only a few terms of (6) will need to be recomputed if x can appear together with a limited number of descriptors.

4. APPLICATION TO CHEMICAL STRUCTURE SEARCHING

Through the use of a method of superimposed coding invented by Mooers[15] for forming bit string screens, we have achieved the ability to weight each descriptor by the assignment of a lesser or greater number of ones to its individual

code. This allows us to grade fragments according to their marginal discrimination in the system. In this section we arrive at an estimate for marginal discrimination which is an especially good approximation for chemical structure information. It should be useful in other fields, also, if they meet certain general requirements.

Our first idea was to approximate the marginal discrimination of each fragment by its first-order discrimination, $-Q(f)$ log $Q(f)$, where $Q(f)$ is its incidence. In that case we would simply assign $-\log Q(f)$ ones to the code of fragment f, rounded off to the nearest integer, and this would give f about $-Q(f)$ log $Q(f)$ ones in the average superimposed code, not counting overlap. This approach has the attractive feature of not requiring joint incidences, which would be cumbersome to compute. However, a more precise approach, which we may call second order, was finally decided upon. Before describing this approach, we should review how fragments are generated and restricted to avoid some types of redundancy.

Fragments were generated step by step according to size from single atom fragments up to 11 atom ring-chain hybrid fragments. At each step the fragments were divided into three classes according to incidence; the somewhat arbitrary cutoff values were 1% and 0.1%. Thus, the upper class had incidence greater than 1%, the middle class occurred between 0.1 and 1% in the file, and the lower class had incidence less than 0.1%.

For the generation of the fragments, a randomized 10% sample of the file was used instead of the entire file. Thus, we were working with roughly 25,000 out of about 250,000 compounds, which allowed a quite good estimate of the actual incidence, at least down to 0.1%.

Our fragment selection is described in detail in ref 1. We used a sequence of iterations, and only fragments which were extensions of upper class, high-incidence fragments were kept. Fragments which were extensions of lower class, low-incidence fragments were purged. After the third step all branching was curtailed, so only chains and rings were collected. After the seventh step, chains too were left out and only fragments which are part ring were generated. These restrictions were necessary to curtail the generation of an excessive number of fragments.

This iterative method of generating fragments was suggested by the aforementioned work of the Sheffield group. It does avoid the extreme kind of redundancy mentioned earlier. However, the very act of producing fragments which are extensions of earlier fragments introduces some redundancy between a fragment and its parents. This prompted us to modify our first idea for approximating the marginal discrimination of each fragment, and to develop a formula to counteract this redundancy.

A typical case to bear in mind may be that of a five-atom chain, considering as its two parents the two four-atom chains contained in it. Call these parents N and Z and denote the fragment itself by W. It is clear that if the incidence of W is not much less than any one of its parents, then W is redundant. Note that the method of generation guarantees only that one of the parents will have high incidence, not necessarily relative to W, and also that no parent will have extremely low incidence. We derive the second-order approximation to marginal discrimination by including this interaction between a fragment and its parents.

We use eq 6 for discrimination and consider the addition of fragment W under the assumption that W will interact strongly with N and Z and negligibly with all other fragments. Then we can write the marginal discrimination (md) according to the last paragraph of the preceding section.

$$\text{md} = -Q(W) \log R(W) - Q(WZ) \log R(WZ) - Q(WN) \log R(WN) - Q(WZN) \log R(WZN) \quad (7)$$

All other $R(A)$ for $W \in A$ are assumed to be about 1.

Note that

$$Q(W) = Q(WZ) = Q(WN) = Q(WZN)$$

Also,

$$R(W) = Q(W)$$

$$R(WZ) = Q(WZ)/R(W)R(Z) = 1/R(Z)$$

$$R(WN) = 1/R(N)$$

$$R(WZN) = Q(WZN)/R(WZ)R(WN)R(ZN)R(W)R(Z)R(N) = 1/R(ZN)$$

Substituting into (7), we get

$$\text{md} = -Q(W) \log Q(W)/R(Z)R(N)R(ZN) = -Q(W) \log Q(W)/Q(ZN) \quad (8)$$

Thus the derived formula 8 properly relates the marginal discrimination of a fragment to the ratio between the incidence of the fragment and the joint incidence of its parents. Now we wish to avoid computing $Q(ZN)$. To do so we use the following heuristic.

Note that $Q(ZN)$ is always smaller than or as small as both $Q(Z)$ and $Q(N)$. In fact, in chemical structures, $Q(ZN)$ is often quite close to the minimum of $Q(Z)$ and $Q(N)$. Call this minimum $\mathbf{Q}(ZN)$. If we use $\mathbf{Q}(ZN)$ as an approximation for $Q(ZN)$, we satisfy the following conditions.

1. We are giving W at least as much weight as given in eq 8; therefore no fragment is deprived of a rightful share according to its marginal discrimination.

2. In going from the first-order approximation for marginal discrimination to the second-order approximation, each fragment is reduced in importance according to the incidence of its parent of lowest incidence. This is a good measure of redundancy and a significant saving in screen bits.

3. In some cases the approximation is very good, such as when one parent has much lower incidence than the others. In the case of a symmetric fragment with only one parent the approximation is exact.

Thus, the number of one bits assigned in the code for each fragment W is $-\log Q(W)$ if W has no parent. Otherwise it is $\log \mathbf{Q}(ZN)/Q(W)$ where $\mathbf{Q}(ZN)$ is the incidence of its parent of lowest incidence. In either case the quantity must be rounded off since only an integral number of one bits can be assigned. In the case that it is assigned zero one bits the fragment is altogether eliminated from the system. When a fragment that has a parent is eliminated, its lowest incidence parent replaces it when its turn comes to be used as a parent. This replacement allows the assignment of one bits to mirror the hierarchical relationships of the fragments finally kept in the system.

5. SUMMARY

An information theoretic approach to descriptor selection has been developed which begins with a definition of discrimination in terms of incidence and mutual incidence of the descriptors. The main theoretical result is a reformulation of the expression for discrimination in terms involving a redundancy factor. The new formulation allows one to easily see the contribution to discrimination of individual descriptors provided they have limited associations with other descriptors. The extreme case of independence shows that the optimal incidence of each descriptor under conditions of independence is $1/e$.[16] In the case of hierarchial descriptors, we can consider the main association as occurring between a descriptor and its most immediate predecessor in the hierarchy. This approximation allows us to estimate the marginal discrimination of a potential new descriptor. Such an estimate is especially

applicable in chemical structure searching. These considerations have permitted the design of an efficient screening system for substructure searching. It seems likely that these methods will prove useful also in other areas than chemistry.

ACKNOWLEDGMENT

I thank Alfred P. Feldman for providing many of the ideas in an intuitive form and also for comments on the manuscript.

APPENDIX

We begin from the definition 1 which we rewrite here

$$D(K) = \Sigma -P(A) \log Q(A) \quad \text{all subsets } A \text{ of } K \tag{A1}$$

First we wish to eliminate the $P(A)$ by expressing them in terms of the $Q(A)$. This can be done by using the inclusion–exclusion principle. From Knuth,[17] the expression for the number of elements in a set of N elements but not in any of the subsets $S_1, S_2, \ldots, S_m$ can be written as follows.

$$N - \sum_{1\le i\le m} |S_i| + \sum_{1\le i<j\le m} |S_i \cap S_j| - \sum_{1\le i<j<k\le m} |S_i \cap S_j \cap S_k| + \ldots + (-1)^m |S_1 \cap \ldots \cap S_m| \tag{A2}$$

We need the sets $\Pi(A)$, $\Psi(A)$, which we recall are, respectively, the set of objects which yield as items precisely A and the set of objects which yield items containing A. The cardinalities of these sets are equal to $nP(A)$ and $nQ(A)$, respectively.

We will also require a notation for subsets and supersets of a set of descriptors A. We let A_i^j stand for a representative superset of A with j more elements than A. j can vary from 0 to $k - a$, where a is the cardinality of A. All the ${}_{k-a}C_j$ supersets of A with j elements added will generally be used together, which means that i will range from 1 to ${}_{k-a}C_j$. In the same spirit we let A_i^{-j} stand for subsets of A with j less elements than A, $1 \le j \le a$. In this case i will range from 1 to ${}_aC_j$.

To find $\Pi(A)$ we must subtract from $\Psi(A)$ all the items A_i^j for $1 \le j \le k - a$ and $1 \le i \le {}_{k-a}C_j$. We can apply (A2) if we use as S_i the set of items $\Psi(A_i^1)$. Note that the intersection of j of these sets is some $\Psi(A_i^j)$ for $1 \le j \le k - a$. So we have

$$|\Pi(A)| = |\Psi(A)| - \sum_{1\le i\le k-a} |\Psi(A_i^1)| + \sum_{1\le i\le {}_{k-a}C_2} |\Psi(A_i^2)| - \sum_{1\le i\le {}_{k-a}C_3} |\Psi(A_i^3)| + \ldots + (-1)^{k-a} |\Psi(K)|$$

This can be written more compactly.

$$|\Pi(A)| = \sum_{0\le j\le k-a} (-1)^j \sum_{1\le i\le {}_{k-a}C_j} |\Psi A_i^j)| \tag{A3}$$

where $A_1{}^0 = A$. Dividing both sides of (A3) by n gives us $P(A)$ in terms of $Q(A)$.

$$P(A) = \sum_{0\le j\le k-a} (-1)^j \sum_{1\le i\le {}_{k-a}C_j} Q(A_i^j) \tag{A4}$$

We can now substitute (A4) into (A1) and rearrange terms, collecting all the log terms for each set A. This follows the rearrangement performed in the earlier examples.

$$D(K) = -\Sigma Q(A) \sum_{0\le j\le a} (-1)^j \sum_{1\le i\le {}_aC_j} \log Q(A_i^{-j}) \tag{A5}$$

Another way to express (A5) simplifies the sum of the logs to the log of a product.

$$D(K) = -\Sigma Q(A) \log \prod_{0\le j\le a} \prod_{1\le i\le {}_aC_j} Q(A_i^{-j})^{(-1)^j} \tag{A6}$$

Comparing (A6) with the statement of the theorem, we see that it is sufficient to show the following for all subsets A of K.

$$R(A) = \sum_{0\le j\le a} \sum_{1\le i\le {}_aC_j} Q(A_i^{-j})^{(-1)^j} \tag{A7}$$

By refering to the definition of $R(A)$, we see that it is expressed inductively in terms of the $Q(A_i^{-j})$. In theory, it is merely necessary to expand the definition and simplify terms. The expansion, however, produces an intermediate form with very high multiplicities of the $Q(A_i^{-j})$, and it is in any case easier to proceed by induction on a, the cardinality of A.

The definition of $R(A)$ can be rewritten with the notation of A_i^{-j}.

$$R(A) = Q(A) \prod_{1\le j\le a} \prod_{1\le i\le {}_aC_j} R(A_i^{-j})^{-1} \tag{A8}$$

The induction hypothesis is that (A7) holds for subsets A of K of cardinality less than a. Thus such an equation holds for all sets A_i^{-j} on the right side of eq A8 for $1 \le j \le a$. We can evaluate each term $R(A_i^{-j})$ for j varying from 1 through a and i varying from 1 through ${}_aC_j$.

$$R(A_i^{-j}) = \prod_{j\le t\le a} \prod_{1\le k\le {}_{a-j}C_{t-j}} Q((A_i^j)_k{}^{-(t-j)})^{(-1)^{(t-j)}} \tag{A9}$$

In fact, each way of placing j descriptors among the t descriptors leads to a contribution from (A9) to be substituted into (A8). The sets $(A_i^{-j})_k{}^{-(t-j)}$ can be written $A_{i+k}{}^{-t}$ and since i varies from 1 through ${}_aC_j$ and k varies from 1 to ${}_{a-j}C_{t-j}$, $i + k$ varies from 1 through $({}_aC_j)({}_{a-j}C_{t-j}) = {}_aC_t$. We can change $i + k$ to k uniformly and enumerate all contributions from (A9) for a single value of j from 1 through a.

$$\prod_{1\le i\le {}_aC_j} R(A_i{}^{-j}) = \prod_{j\le t\le a} \prod_{1\le k\le {}_aC_t} Q(A_k{}^{-t})^{(-1)^{(t-j)}{}_tC_j} \tag{A10}$$

Finally we can combine the terms for j from 1 through a by substituting from (A10) into (A8).

$$R(A) = Q(A) \prod_{1\le j\le a} \prod_{j\le t\le a} \prod_{1\le k\le {}_aC_t} (Q(A_k{}^{-t})^{(-1)^{(t-j)}{}_tC_j})^{-1} \tag{A11}$$

Reordering terms we get

$$R(A) = Q(A) \prod_{1\le t\le a} \prod_{1\le k\le {}_aC_t} Q(A_k{}^{-t}) \sum_{1\le j\le t} (-1)^{(t-j-1)}{}_tC_j \tag{A12}$$

The following identity derives from the expansion of $(1\text{-}1)^t$.

$$\sum_{1\le j\le t} (-1)^j {}_tC_j = -1$$

Therefore the exponent summation in (A12) works out to be $(-1)^t$. Noticing that $Q(A)$ can be written $Q(A_1{}^0)$ gives us the equivalent of (A7) and completes the proof.

REFERENCES AND NOTES

(1) A. Feldman and L. Hodes, "An Efficient Design for Chemical Structure Searching. I. The Screens", *J. Chem. Inf. Comput. Sci.*, **15**, 147–52 (1975).
(2) W. S. Meisel, "Computer-Oriented Approaches to Pattern Recognition", Academic Press, New York and London, 1972.
(3) N. Jardine and R. Sibson, "Mathematical Taxonomy", Wiley, London, 1971.
(4) G. Salton et al., "A Theory of Term Importance in Automatic Text Analysis", *J. Am. Soc. Inf. Sci.*, **26**, 33–44 (1975).
(5) A. Bookstein and D. R. Swanson, "A Decision Theoretic Foundation for Indexing", *J. Am. Soc. Inf. Sci.*, **26**, 45–50 (1975).
(6) R. C-T. Lee, "Application of Information Theory to Select Relevant Variables", *Math. Biosci.*, **11**, 153–61 (1971).
(7) J. Kryspin and A. Norwich, "Application of Information Calculus to Medical Data Analysis and Reduction", *Math. Biosci.*, **17**, 165–72 (1973).

(8) J. Lederberg, et al., "Applications of Artificial Intelligence for Chemical Inference. I. The Number of Possible Organic Compounds. Acyclic Structures Containing C, H, O, and N", *J. Am. Chem. Soc.*, **91**, 2973–76 (1969).
(9) M. Milne, et al., "Search of CA Registry (1.25 Million Compounds) with the Topological Screens System", *J. Chem. Doc.*, **12**, 183–9 (1972).
(10) D. Lefkovitz, "The Large Data Base File Structure Dilemma", *J. Chem. Inf. Comput. Sci.*, **15**, 14–9 (1975).
(11) M. F. Lynch, et al., "Computer Handling of Chemical Information", McDonald, London, and American Elsevier, New York, 1971, p 84.
(12) D. J. Gluck, "A Chemical Structure, Storate and Search System Development at DuPont", *J. Chem. Doc.*, **5**, 43–51 (1965).
(13) Reference 11, p 91.
(14) G. W. Adamson, et al., "Strategic Considerations in the Design of a Screening System for Substructure Searches on Chemical Structure Files", *J. Chem. Doc.*, **13**, 153–7 (1973).
(15) C. N. Mooers, "Zatocoding Applied to the Mechanical Organization of Knowledge", *Am. Doc.*, **2**, 20–32 (1951).
(16) If we were dealing with binary variables instead of descriptors, as differentiated in the opening paragraphs, then the optimal incidence would be $1/2$ instead of $1/e$.
(17) D. E. Knuth, "The Art of Computer Programming. Vol. I. Fundamental Algorithms", Addison-Wesley, Reading, Mass., 1968, p 179.

Experimental Algorithmic Generation of Articulated Index Entries from Natural Language Phrases at Chemical Abstracts Service†

STANLEY M. COHEN*, DAVID L. DAYTON, and RICARDO SALVADOR**

Chemical Abstracts Service, The Ohio State University, Columbus, Ohio 43210

Received November 17, 1975

An algorithm was developed which transforms coded, natural language phrases into multiple index entries consisting of headings and articulated subordinate phrases. Coding and dictation procedures were developed to allow document analysts to input phrases and associated data in a format compatible with the Chemical Abstracts Service (CAS) index production system. In an experiment, 13 CAS document analysts generated, coded, and dictated phrases descriptive of the content of documents input to the CAS processing stream. These phrases were then transformed via the articulation algorithm into entries of the type used in *Chemical Abstracts* (CA) volume indexes. In an evaluation of over 20,000 algorithm-articulated entries, 97.2% were judged intelligible and acceptable. The input of phrases required 62.0% of the keystrokes required for the input of individual entries. The major problem was the error level in the analyst-dictated, clerically keyboarded codes for input phrases. On-line interactive processing techniques may essentially eliminate this problem. The phrase input procedures are being adapted for on-line experimentation.

INTRODUCTION

At present production levels, over 2,000,000 index entries are published yearly in the *Chemical Abstracts* (CA) Chemical Substance and General Subject Indexes. To produce these index entries, document analysts extract information from original documents and/or abstracts and then organize this information into a variety of data components used in the Chemical Abstracts Service (CAS) index production system. Two of the components of the CA volume index entries are of primary interest in this study:

1. *Heading*: the primary access term for an index entry and the basis for alphabetical arrangement of the entries within the index
2. *Text modification*: the indented, subordinate phrase modifying the heading.

Figure 1 illustrates a typical CAS index entry consisting of *heading* and *text modification*.

In the present index production system, document analysts identify and dictate all the data components for each index entry individually. Then clerical staff keyboard the dictated entries for input to the system. Most documents indexed require multiple index entries, which are often permutations of the same words and ideas. Figure 2 illustrates such a group of index entries for one document. Notice that the heading portion of one index entry becomes a part of the text modification of other entries.

Armitage and Lynch[1,2] have described a process called articulation which algorithmically generates multiple index entries from a single, descriptive, title-like phrase. Figure 3 illustrates a title-like phrase which, when subjected to an articulation algorithm, could produce the set of index entries in Figure 2. When a single input phrase can be successfully transformed into a set of index entries, one can readily see the potential for reducing the effort in formulating and dictating multiple permutations of the index entry words as well as the potential for reducing the keyboarding effort required for input. Given the magnitude of the CA volume indexes, this potential reduction of input effort represents substantial savings. Reduced input requirements would take on added importance in any future on-line environment where the document analyst would do his own keyboarding at a terminal.

This paper describes a study of phrase input for indexing use at CAS. Each of the four parts discusses one of the four main aspects of the study.

Part I. An articulation *algorithm* capable of transforming analyst-dictated natural language phrases into multiple index entries of the syntactical style of CA volume index entries.

Part II. Procedures for coding natural language phrases suitable for algorithmic articulation. These *coding procedures* were designed to generate articulated index entries and associated data in data element format compatible with the CAS index production system.

Part III. *Testing* the articulation algorithm and the coding procedures in a production-like environment.

Part IV. An evaluation of the *results* of the study.

† Presented before the Division of Chemical Information, 170th National Meeting of the American Chemical Society, Chicago, Ill., Aug 27, 1975.

* Author to whom correspondence should be addressed.
** Servicio de Marketing, Compañía Telefónica Nacional de España, Madrid 20, Spain.