NOTICE: When government or other drawings, specifications or other data are used for any purpose other than in connection with a definitely related government procurement operation, the U. S. Government thereby incurs no responsibility, nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use or sell any patented invention that may in any way be related thereto.

63-3-3

SP-1153

COMPUTER GENERATION OF WORD ASSOCIATION MAPS FOR MAN-MACHINE COMMUNICATION

Cameron Watson

25 March 1963

(SP Series)

SP-1153

COMPUTER GENERATION OF WORD ASSOCIATION MAPS FOR MAN-MACHINE COMMUNICATION

by

Cameron Watson

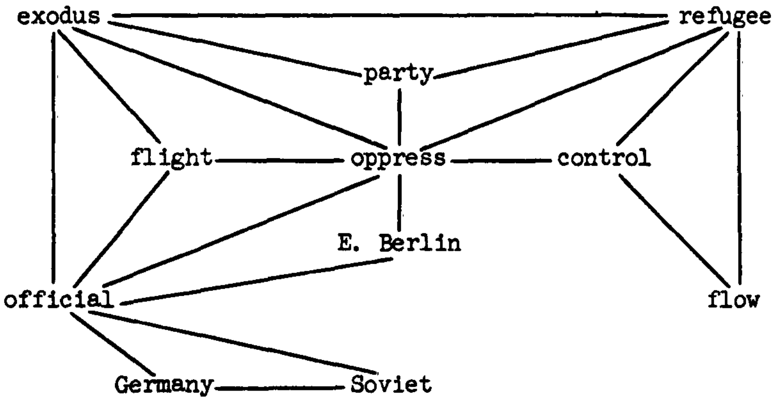
25 March 1963

SYSTEM DEVELOPMENT CORPORATION, SANTA MONICA, CALIFORNIA

INTRODUCTION

The modern digital computer is much like an incredibly swift blind man, capable of handling information only by running a deft finger along a one-dimensional line of data. The human being is much slower at this kind of work, but he has other talents which can reduce the difference in speed. Man has the ability to grasp and integrate information displayed in two dimensions with great speed and agility, an ability epitomized in the cliche, "A picture is worth ten thousand words."

Lauren Doyle must have had this in mind when he conceived the association map as a method of man-machine communication in an information retrieval system.[1] Such a map may be considered a graphic summary of the contents of a file or library. Highly associated words drawn from the texts contained in the file are printed on a page such that associations between words are indicated by interconnecting lines. The following example is drawn from a larger display generated from intelligence reports on the Berlin situation. Even from this limited map one may obtain some idea of the contents of the file.



[1] Doyle, Lauren B., "Indexing and Abstracting by Association, Part I," SP-718/001/00, System Development Corporation, April 1962.

Associations may be defined and measured in many ways. Within the context of this paper association will be limited to word pairs. Two words are defined as associated if they appear together in a sentence, paragraph, abstract or whole document, or are index terms assigned to a paragraph, abstract or document. Given word strings of defined length, maps may be generated for parts or representations of documents, whole documents or libraries of documents.

The choice of string length -- sentence, paragraph, document -- is dependent upon the level of specificity or generality desired, frequency parameters of word occurrence, the method of computation and formula characteristics. Association coefficients measuring degrees of association between words provide the basis for drawing maps and estimating the relative importance of the diverse associations.

The promise of association mapping as a psychological tool for browsing through or searching an automated information store may be illustrated by assuming an operating system applied to intelligence data. The intelligence analyst requests the display of a general map of the store in a given subject area. From the map he selects one or more words delimiting his specific subject interest and inputs them to the computer. The computer accepts the input words as building blocks for a new map. Searching the library, it finds words which have high association with those provided by the analyst and thus narrows the remaining search to a subset of texts within the library. The new map is then computed and displayed.

The analyst examines the new map and further narrows the search by selecting additional words of interest. New maps are generated as needed, culminating in display of single-document maps, permitting rejection of those which are related but peripheral to the analyst's interest. The display of abstracts, whole text or furnishing of hard copy may then be implemented.

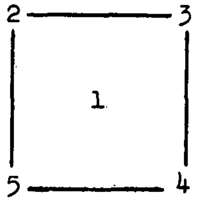
The mapping procedure exhibits the subject matter both of a set of documents and of individual documents by the same method. It shows associations which might not otherwise have been suspected, thus indicating alternative retrieval paths. The measures of association are chosen to give greater weight to words contained in and related to search requests than those in the general store. The search may be restructured to conform to the user's net of inferences at any point in a sequence of mapping iterations.

DISCUSSION

We have used the term "association map" to denote a two-dimensional array of elements with lines drawn between the elements to indicate an associative relationship. We shall now consider the problems of generating and displaying such associations through the use of a digital computer.

The first problem we face arises from the restriction that the map be two-dimensional. A large number of highly correlated words results in a tangle of interconnecting lines. Since a primary objective of an association map is rapid communication of information to a human being, it may be most useful to restrict the map display to the illustration of the most important relationships with few or no association lines crossing. Let us consider a "primitive map" containing five elements with all possible connections shown:

Primitive Map



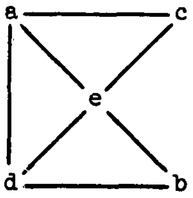
The given set of words are labelled a, b, c, d, e and their associations as computed by a given formula might be:

Association Table

	a	b	c	d	e
a)	.0	.0	.1	.2	.3
b)	.0	.0	.0	.1	.4
c)	.1	.0	.0	.0	.1
d)	.2	.1	.0	.0	.2
e)	.3	.4	.1	.2	.0

In this case, the assignment can be made quite easily, and an acceptable output map would be:

Association Map



In this example all associations can be indicated by connecting lines. Since in general this will not be the case, we need a way to measure the "goodness" of our assignments. One obvious measure is the sum of the coefficients represented by lines on the map.

To formulate the assignment problem more rigorously, let:

$C = (c_{ij}) =$ correlation matrix for words;

$D = (d_{ij}) =$ correlation matrix for map points;

$X = (x_{ij}) =$ assignment matrix.

The correlation matrix for words corresponds to the association table in the previous example. In general, the matrix need not be symmetric; that is, c_{ij} need not equal c_{ji}.

The correlation matrix for map points is analogous to the correlation matrix for words except that the d_{ij} will be one or zero -- a line can or cannot be drawn between point i and point j. This matrix is symmetric.

The assignment matrix is a binary matrix associating word i with point j. Values will be 1 or 0. Generally the matrix will be non-symmetric and will have the property such that any row or column contains only one non-zero entry. This corresponds to the isomorphic character of the map. Each word is assigned to one and only one point; each point has one and only one word assigned to it.

The assignment problem here is the maximization of:

$$\sum_{i,j} c_{ij} r_{ij},$$

where $(r_{ij}) = XDX^t$,

an optimal assignment problem with a varying rating matrix (being a function of the remaining assignments). A solution through linear programming might involve n! assignments for an exhaustive search which is obviously impracticable. An alternative is the selection of an algorithm which will arrive at a relative maximum without excessive computation.

Such an algorithm was sought and found as follows:

1. Assign words 1, 2, ..., n to map points 1, 2, ..., n;
2. Evaluate the sum,

$$\sum_{\substack{i=1\\ i\neq j,\,k\\ \text{for all } k>j\,=\,1,\,2,\,\ldots,\,n-1}}^{n} (c_{ji}d_{ji} - c_{ji}d_{ki} + c_{ki}d_{ki} - c_{ki}d_{ji})$$

At each step if the sum is negative, rows j and k and columns j and k are interchanged in both C and X;

3. Repeat until no negative values are encountered.

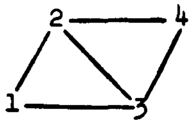
To illustrate this procedure, consider the following example, given four words, their association coefficients and a primitive map.

Words

a) fifteen
b) four
c) ten
d) fourteen

$$C = \begin{array}{ccccc} & \underline{a} & \underline{b} & \underline{c} & \underline{d} \\ a) & .0 & .1 & .2 & .3 \\ b) & .1 & .0 & .0 & .1 \\ c) & .2 & .0 & .0 & .2 \\ d) & .3 & .1 & .2 & .0 \end{array}$$

Map



$$D = \begin{array}{ccccc} & \underline{1} & \underline{2} & \underline{3} & \underline{4} \\ 1) & 0 & 1 & 1 & 0 \\ 2) & 1 & 0 & 1 & 1 \\ 3) & 1 & 1 & 0 & 1 \\ 4) & 0 & 1 & 1 & 0 \end{array}$$

Assignments

1)	a
2)	b
3)	c
4)	d

NOTE: For computation, it is easier to store an assignment table. The matrix is a mathematical convenience.

For j = 1 and k = 2, our sum is:

$$\begin{aligned} \text{sum: } jk &= c_{13}d_{13} - c_{13}d_{23} + c_{23}d_{23} - c_{23}d_{13} \\ &+ c_{14}d_{14} - c_{14}d_{24} + c_{24}d_{24} - c_{24}d_{14} \\ &= .2\ (1) - .2\ (1) + .0\ (1) - .0(1) \\ &+ .3\ (0) - .3\ (1) + .1\ (1) - .1\ (0) \\ &= .2 - .2 - .3 + .1 = -\ .2 \end{aligned}$$

The sum is negative; so we interchange rows 1 and 2, and columns 1 and 2 of matrix C. The assignment-table entries 1 and 2 must also be exchanged:

Assignments	
1)	b
2)	a
3)	c
4)	d

$$C = \begin{array}{lcccc} & \underline{b} & \underline{a} & \underline{c} & \underline{d} \\ b) & .0 & .1 & .0 & .1 \\ a) & .1 & .0 & .2 & .3 \\ c) & .0 & .2 & .0 & .2 \\ d) & .1 & .3 & .2 & .0 \end{array}$$

For $j = 1$, $k = 3$, we have:

$$\text{sum}_{jk} = .1 - .1 + .2 - .2 + 0 - .1 + .2 - 0 = +.1$$

Since the sum is positive, no interchanges will be made. Continuing:

$\text{sum}_{14} = 0$

$\text{sum}_{23} = 0$

$\text{sum}_{24} = 0$

$\text{sum}_{34} = -.1$

We must interchange rows 3 and 4, columns 3 and 4, and assignment-table entries 3 and 4:

Assignments	
1)	b
2)	a
3)	d
4)	c

$$\begin{array}{lcccc} & \underline{b} & \underline{a} & \underline{d} & \underline{c} \\ b) & 0 & .1 & .1 & 0 \\ a) & .1 & 0 & .3 & .2 \\ d) & .1 & .3 & 0 & .2 \\ c) & 0 & .2 & .2 & 0 \end{array}$$

Beginning again with $j = 1$, $k = 3$:

$\text{sum}_{12} = +.2$

$\text{sum}_{13} = +.2$

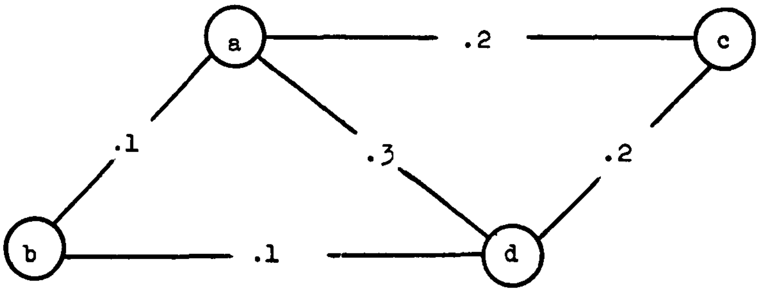
$\text{sum}_{14} = 0$

$$\mathrm{sum}_{23} = 0$$

$$\mathrm{sum}_{24} = +.1$$

$$\mathrm{sum}_{34} = +.1$$

Since all of the sums are positive on the second pass, no improvements are possible and our final map appears thus:



* * *

An important area remaining for discussion is the selection of words for display on the map. On the rather precarious assumption that the most highly associated or intercorrelated terms will give a good picture of the library contents, we can use these for the initial map. But how do we select key words once the user has specified several inputs as descriptive of his search requirement? One method is best explained by an example.

Suppose the user selects three words -- call them X, Y, and Z -- as lying in his area of interest. These words are correlated with a, b, ..., j as shown in the table:

	X	Y	Z
a	.5	.5	.5
b	.5	.5	.0
c	.5	.0	.0
d	.0	.0	.0
e	.1	.1	.1
f	.1	.1	.0
g	.1	.0	.0
h	.5	.5	.1
i	.5	.1	.1
j	.5	.1	.0

Our problem is to select from the table above those words which are most likely to be of interest to the user. To do this, we need some sort of function which assigns a "desirability index" to each word. Consider

$$V = \left[\max(p,s)\right] \cdot \left[\max(q,s)\right] \cdot \max(r,s)$$

where p = the correlation between a keyword and X;
q = the correlation between the same word and Y;
r = the correlation between the same word and Z;
s = a small constant added to avoid zero terms.

In the above example, we let $s = 0.01$; in this case, we obtain, in order of "desirability":

	X	Y	Z	V
a)	.5	.5	.5	.125000
h)	.5	.5	.1	.025000
i)	.5	.1	.1	.005000
b)	.5	.5	.0	.002500
e)	.1	.1	.1	.001000
j)	.5	.1	.0	.000500
f)	.1	.1	.0	.000100
c)	.5	.0	.0	.000050
g)	.1	.0	.0	.000010
d)	.0	.0	.0	.000001

By using X, Y, Z, and the highest words in a list such as this, we obtain terms which are closely correlated with the keywords and hence lie nearest to (and hopefully within) the area of interest as specified by the user.

IMPLEMENTATION

As the ideas in the Discussion section evolved, programming for the IBM 7090 was initiated. A 2500-instruction program in SCAT was written, debugged and checked out after several months of effort.

The program is organized in two independent sections. The first section, called "Table-Maker," produces an association tape from index-word or whole text input. For index-word input, document numbers are assigned to each set of index terms. With whole text input, document numbers are assigned to word-strings of arbitrary length, such as sentences, paragraphs or whole documents at the option of the user. Provision is made for elimination of selected words through matching with an exclusion list. Except for exclusions, words contained within a string are treated as sets of index terms. Operating upon any reasonable association formula, Table-Maker finds the "n" words which are most highly associated with the most frequent word and dumps them onto the association tape. It repeats the calculation for the second most frequent word, etc., creating a table for each index word.

The second section of the program, termed "Assignment Optimizer," operates upon the association tape and a second tape containing "keyword" input, and prints out word lists in a specified order, thus furnishing information necessary for drawing maps. The keyword tape provides for the generation of maps correlating the association tape file content with pre-selected terms. The input routine reads a record from the keyword tape; if it is a zero, the first map will reflect a computation upon the "n" most frequent words as index terms. If it is other than zero, the record is interpreted as the number of keywords to be used for the first map. These words are read in from the keyword tape and their tables entered for the value-function computation. The "n" words with the highest values are used to produce the association matrix for the Optimizer.

With initiation of the optimization procedure, the input routine is able to begin bringing in keywords and tables for the next map. When the Optimizer has completed work on the first map, the data flows to the output routine, freeing the Optimizer for the next assignment problem. Thus simultaneous read, write and compute are effected.

Consideration was given to adding a routine to the Assignment Optimizer for the printing out of complete maps using x's or other such characters to indicate association lines, and for providing for a CRT display. The provision of such a capability was deferred because it was clear that further test and experimentation were required for the selection of formulas and an optimum map geometry.

Running times for the two sections depend upon the formulas used, the length of text processed and the number of maps desired for a given text. In a typical experiment, a printout for five maps using a 3000-word text input required a total of 5.8 minutes of 7090 time. The Table-Maker section took 4.6 minutes to produce an association tape containing some 700 tables. The Assignment Optimizer section used 1.2 minutes for generation of the five maps.

At the present the program is capable of accepting a total of 12,000 word occurrences of 4,000 word types for a given computation. It is estimated that the production of five maps for such a total would currently require about twenty-four minutes of machine time. It is expected, however, that a small modification which is to be made in the Table-Maker will reduce this requirement by one-half or more.

EXPERIMENTS AND RESULTS

Three very different kinds of text were used in a series of experiments designed to test the program under development and to explore computational alternatives and various display geometries. The first experimental corpus consisted of one hundred intelligence reports on the Berlin situation. These are short summaries covering a variety of subjects. Each report was assigned twelve index terms which were used as the basis for calculations leading to map generation. Manually produced maps were available for comparison.

Because of ambiguities of interpretation inherent in the intelligence material a second corpus was built from what might be termed "a thesaurus net." Three words, "mean," "run," and "good," were chosen, and synonym chains were built for each through the use of Roget's Thesaurus. Each word and its synonyms were considered as index words for a document. All synonymous words were then listed together with their particular synonyms, each synonymous cluster being treated as a document. This procedure was pursued for synonyms of synonyms thus creating a document file whose derivation and implications were relatively controlled and clear. Each inflected form of a word heading a synonym cluster defined distinct documents for its various meanings as a noun, verb, or adjective. The corpus proved most helpful in program debugging and checkout, and in demonstrating the effects of different computational procedures and formulas.

A third corpus consisted of a key-punched text, totalling 3,000 words drawn from two technical encyclopedia articles on semiconductors and transistors. Experiments were performed on the effects of varying word string length, formulas and selective exclusion of frequent terms.

Two formulas were used for most of the experiments in generating association coefficients:

$$C_{ab} = \frac{F}{A + B - F}, \quad \text{and} \quad C_{ab} = \frac{F^2}{A^2 + B^2}$$

where C_{ab} = association coefficient for words A and B;

A = the number of documents containing or indexed by word A;
B = the number of documents containing or indexed by word B;
F = the number of documents containing or indexed by both A and B.

The two formulas have in common a range from 0 to 1 and are at a maximum when A = B = F. The second formula discriminates more severely between words of differing occurrence frequency than the first, and therefore produces somewhat different maps.

Map Numbers 1, 2 and 3 were generated from a calculation on the thesaurus net based upon the words "mean," "run," and "good." Map Number 1 is a "general map" reflecting the characteristics of the entire file. It is divided into six clusters of varying size, reflecting in part the size of the file. The thesaurus chain resulting from the word "mean" was nearly twice the length of those based on "run" and "good." The isolation of clusters due to the limitation of crossing lines is demonstrated by those appearing at the top and bottom of the display, containing the words "good," "edge," and "right."

Map Nos. 2 and 3 illustrate the differences between two formulas based upon the association of two keywords with the thesaurus net -- "good" and "bad." Clearly the two keywords form distinct sets of associations.

Map Nos. 4, 5, and 6 were developed from intelligence reports of the 1950's describing events related to Berlin. Map No. 4 is a "general map" illustrating the most densely correlated terms from the entire file. Map Nos. 5 and 6 show associations generated from keywords "refugee" and "industrial," respectively. This sequence was chosen to illustrate what might occur in a search. "Refugee" appears on the general map; "industrial" does not. "Industrial" appears on Map No. 5 generated from "refugee" as a keyword, associated with "World War II," "population" and "manpower," and indirectly with "deplete" and "refugee." Another suggestive cluster on Map No. 5 is that appearing in the lower right-hand corner, associating "refugee" with "flow," "May Day," "opportunity," "confusion," "border" and "holiday."

Map Nos. 7 through 12 were produced from unedited text drawn from two technical encyclopedia articles on semiconductors and transistors. Map No. 7 is a general map of the semiconductor article with no words excluded. Map No. 8 is drawn from the same file using the keyword "compensated." Map No. 9 is identical except for the use of an exclusion list of frequently occurring words. All the remaining maps use the same exclusion list:

(1) all one- and two-letter words;
(2) and, are, had, has, its, the, was, been, have, that, this, were, will, their, these, and those.

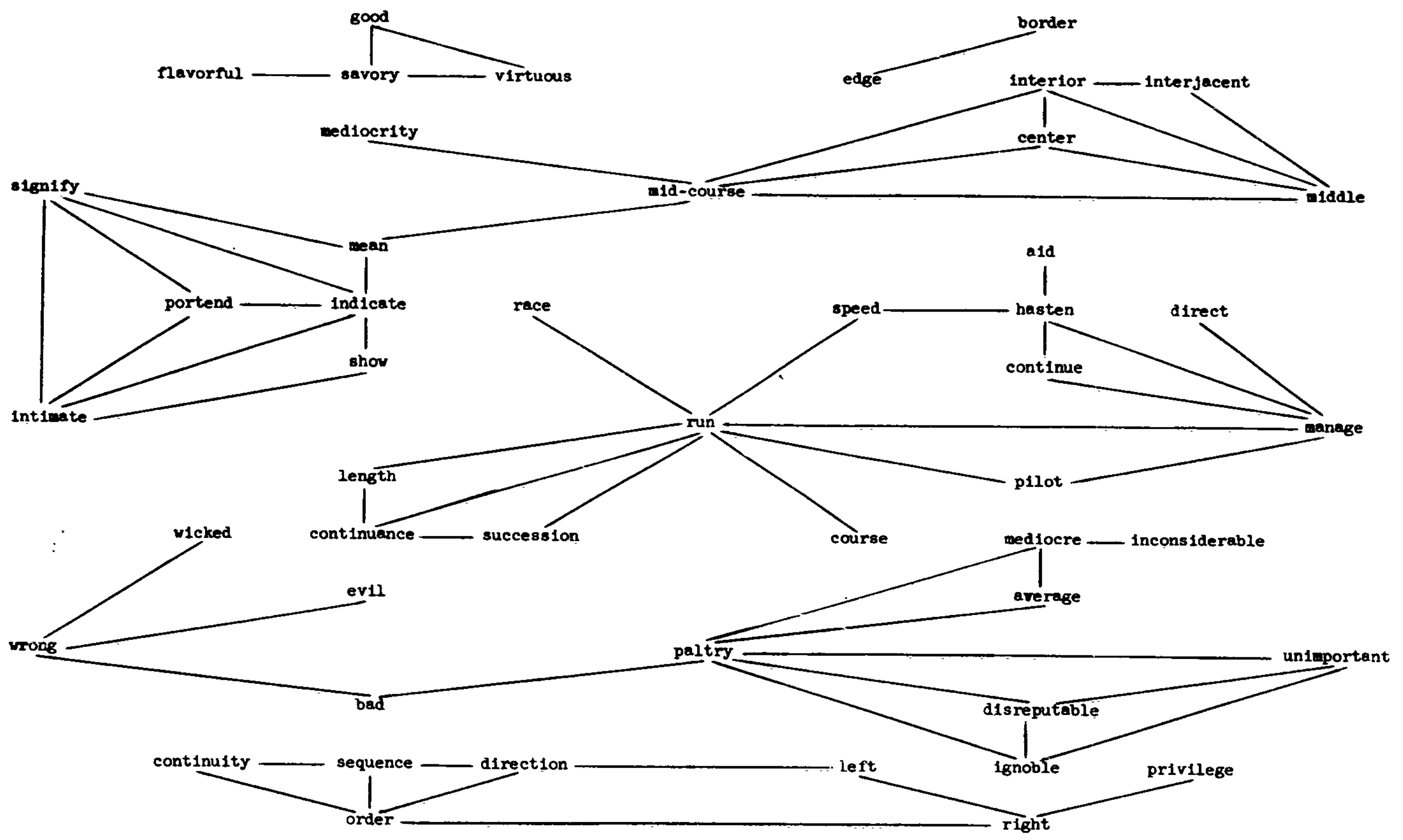
Map No. 10 is again a general map of the semiconductor article with words on the foregoing list excluded. Map No. 11 is a general map of the two articles on semiconductors and transistors combined in one 3000-word file. Map No. 12 exhibits the associations of the keyword "semiconductor" with the file containing the two articles.

The full-text Map Nos. 7-12 used the sentence as the unit of string length for the calculation of coefficients. In light of other experiments applying different string lengths the sentence appears to be the optimum unit for textual association using ordinary prose. The technique might be improved if the independent clause, rather the whole sentence, were defined as the unit, particularly for prose styles which involve frequent use of extended sentences containing many distinct statements. Moreover, the question of so-called "function word" exclusion is one which demands further exploration. The choice of formula has a great effect on relative values of high and low frequency word coefficients. A particular formula may emphasize either.

In the course of this inquiry many map display geometries were tried and found wanting. For clarity of presentation a single map layout providing a maximum of forty-nine word points was chosen for the map examples that follow. The display provides for no crossing of association lines and shows the forty-nine or less most densely associated words based upon an optimization calculation on a 72 x 72 matrix of association coefficients.

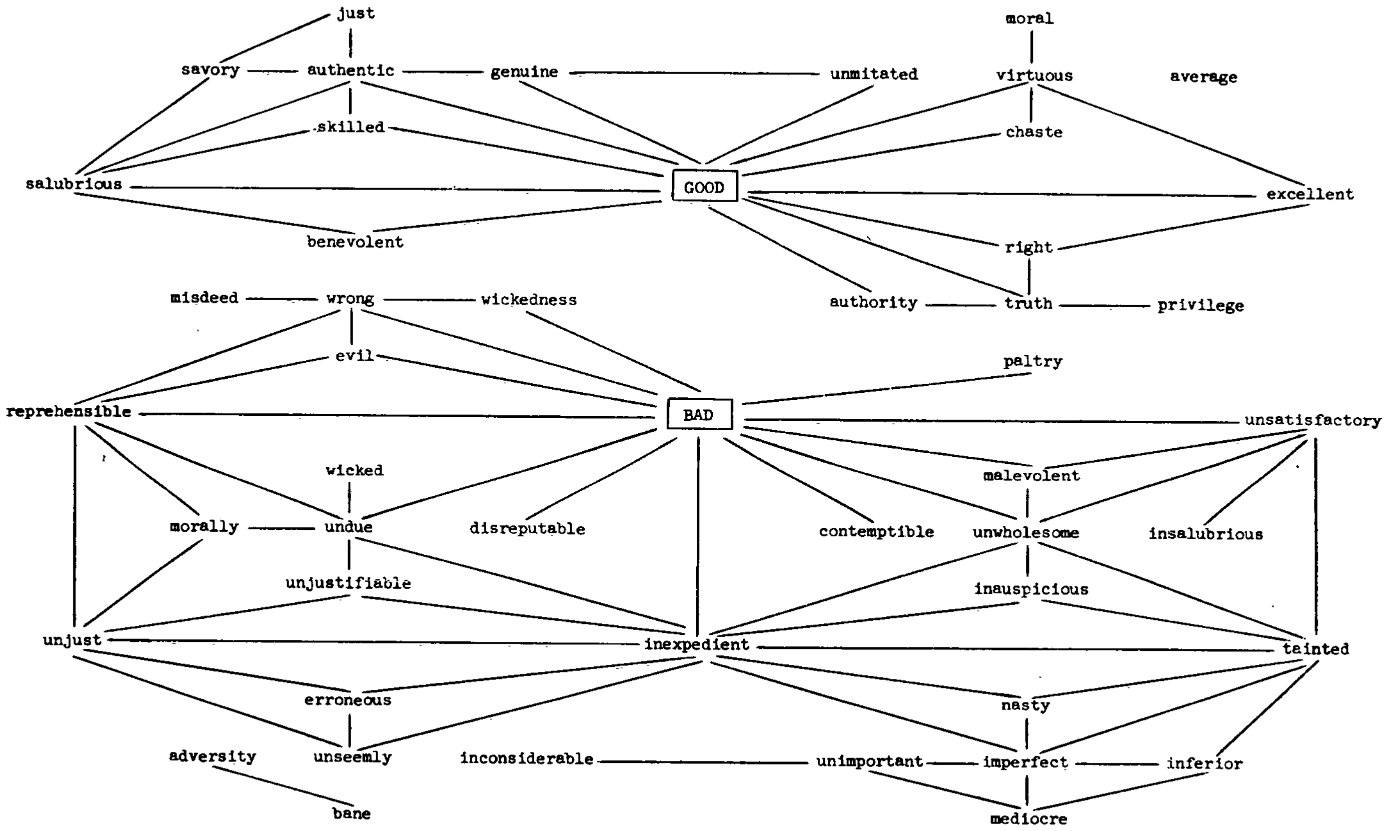
The limitation on crossing lines and on the number of word points which may be shown often reduces the number of associations which may be displayed. This results in the appearance of isolated words or clusters which are associated but not connected with other words displayed or with words whose coefficients were too low to enter the display.

Our experience thus far has indicated that associative mapping is a most promising, even if embryonic, technique. It is obvious also that more developmental work is required before the procedure can be regarded as a practical tool for information retrieval. Work is particularly needed in formula development and map formatting. The map examples illustrate both the promise and the problems of this approach.

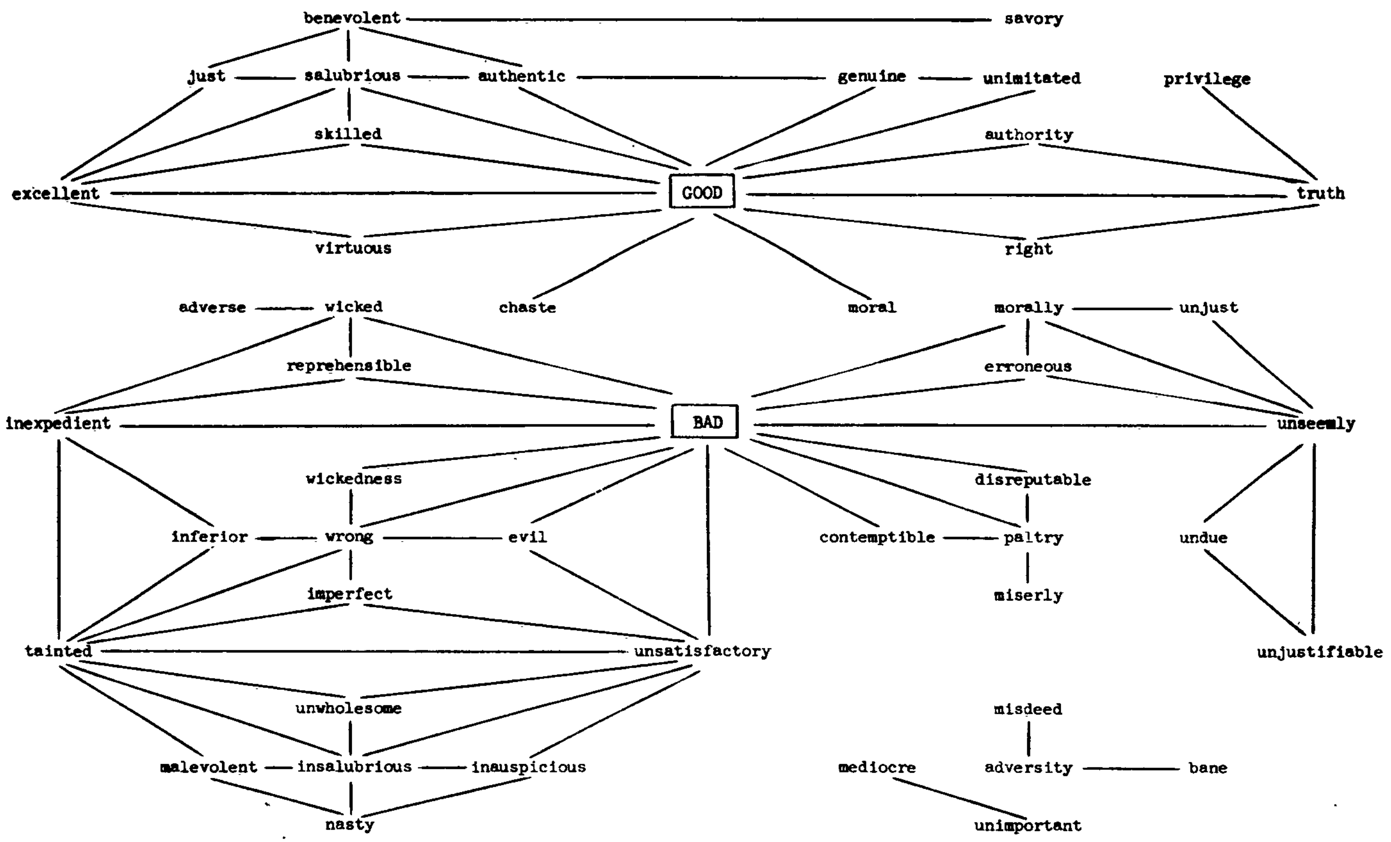


Map No. 1. General Map of Thesaurus Net

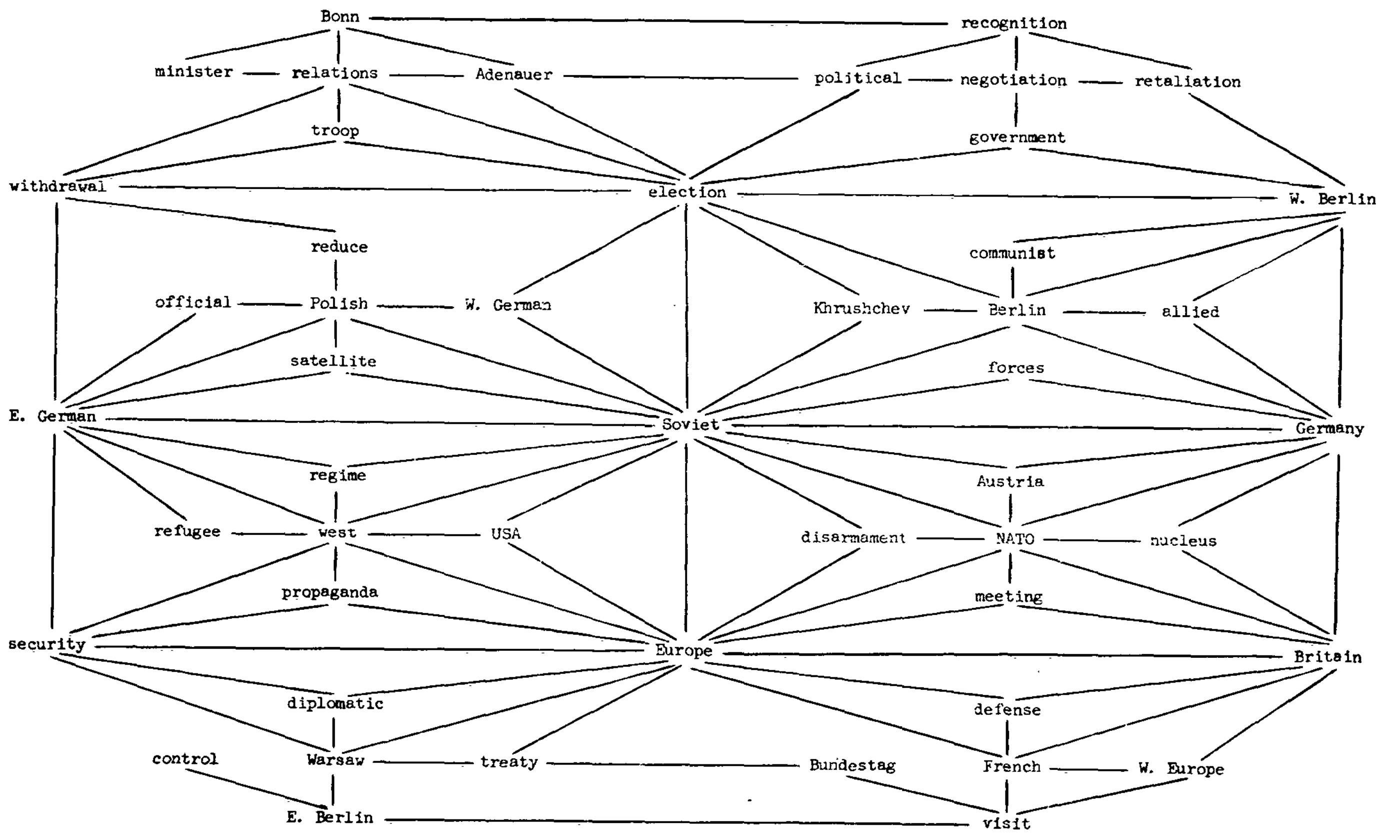
Formula: $C_{ab} = \frac{F}{A + B - F}$



Map No. 2. Thesaurus Map Based on Keywords "Good" and "Bad"   Formula: $C_{ab} = \frac{F}{A + B - F}$

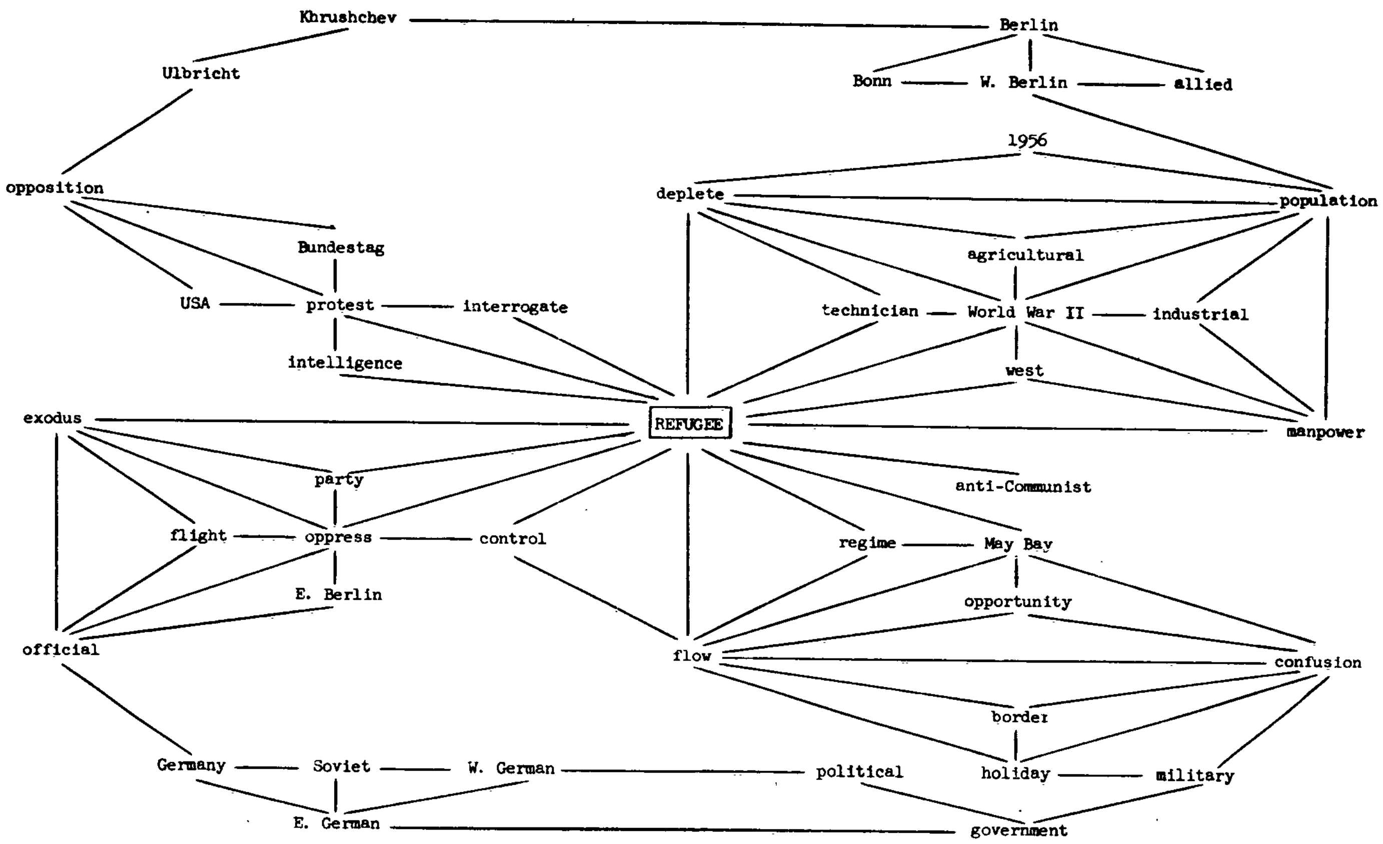


Map No. 3. Thesaurus Map Based on Keywords "Good" and "Bad" Formula: $C_{ab} = \frac{2F^2}{A^2 + B^2}$



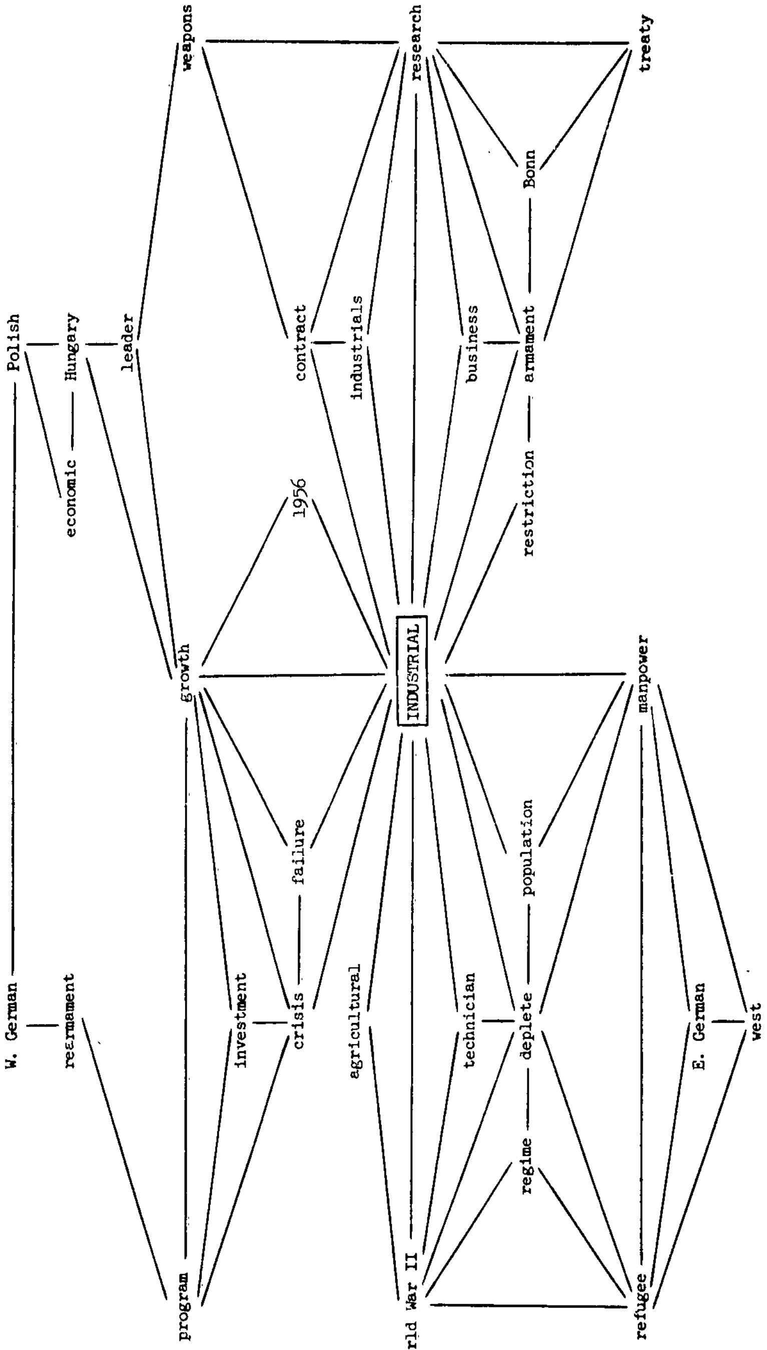
Map No. 4. General Map of Intelligence Reports on Berlin

Formula: $C_{ab} = \frac{F}{A + B - F}$

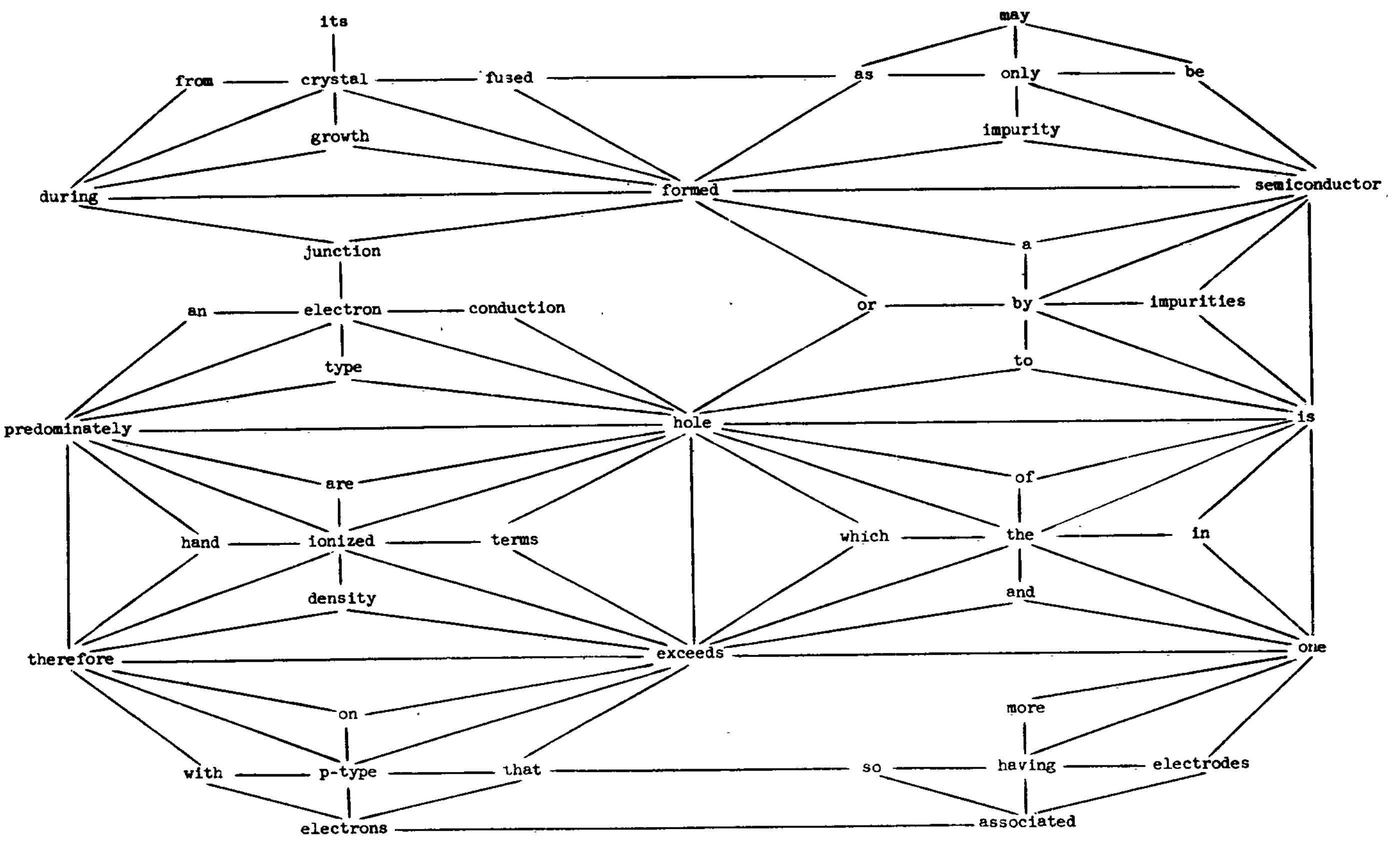


Map No. 5. Intelligence Map Based on Keyword "Refugee"

Formula: $C_{ab} = \frac{F}{A + B - F}$

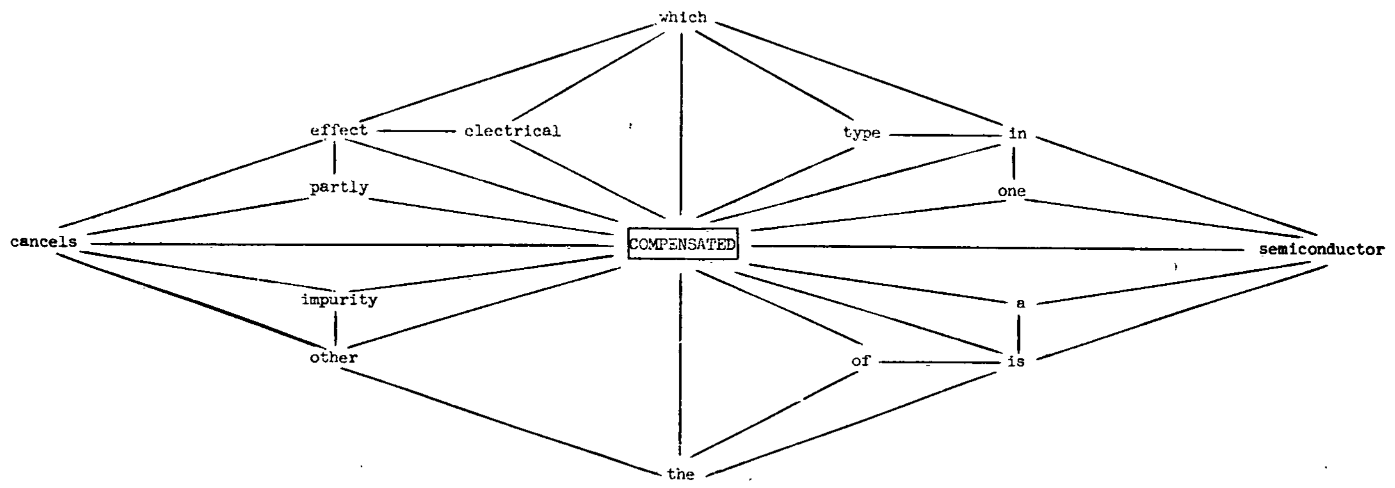


Map No. 6. Intelligence Map Based on Keyword "Industrial" Formula: $C_{ab} = \frac{F}{A + B - F}$

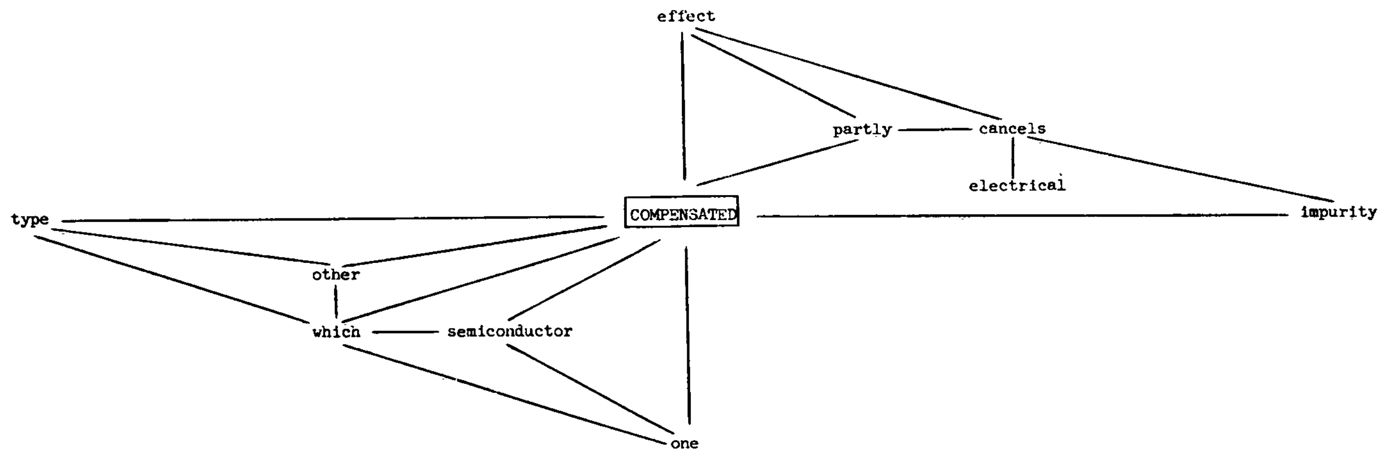


Map No. 7 General Map of Article on Semiconductors

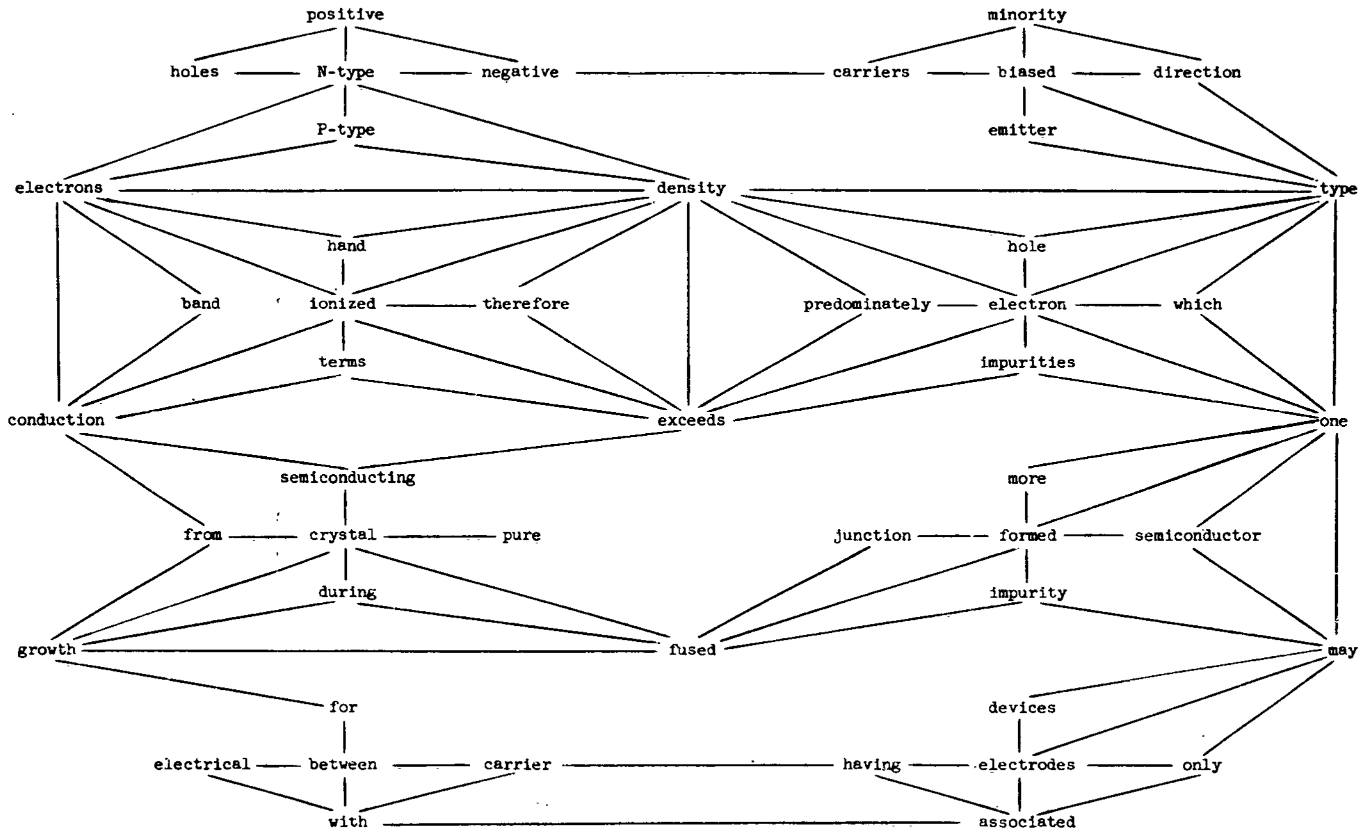
Formula: $C_{ab} = \frac{F}{A + B - F}$



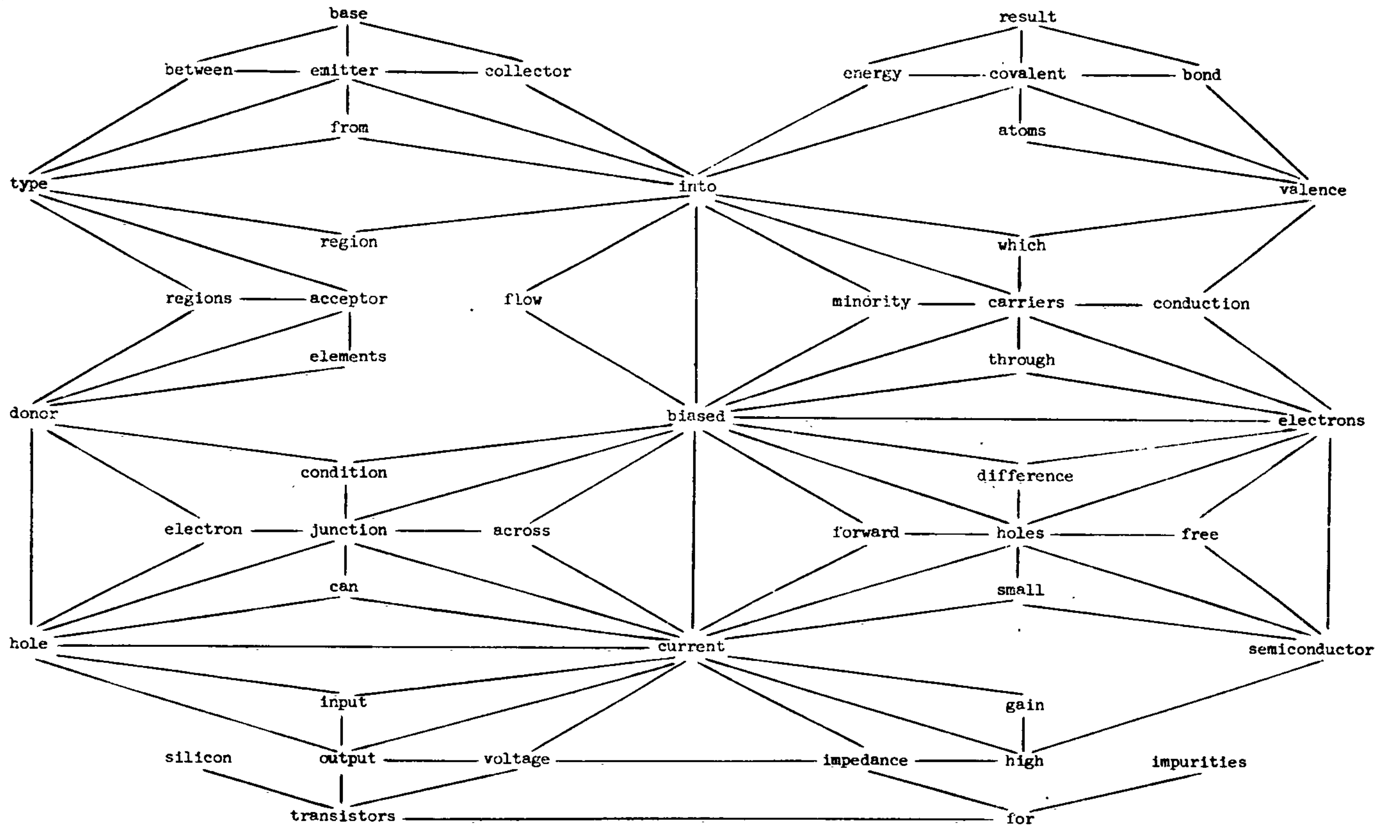
Map No. 8 Semiconductors Map Based on Keyword "Compensated" Formula: $C_{ab} = \frac{F}{A + B - F}$



Map No. 9. Semiconductors Map Based on Keyword "Compensated" Formula: $C_{ab} = \frac{F}{A + B - F}$

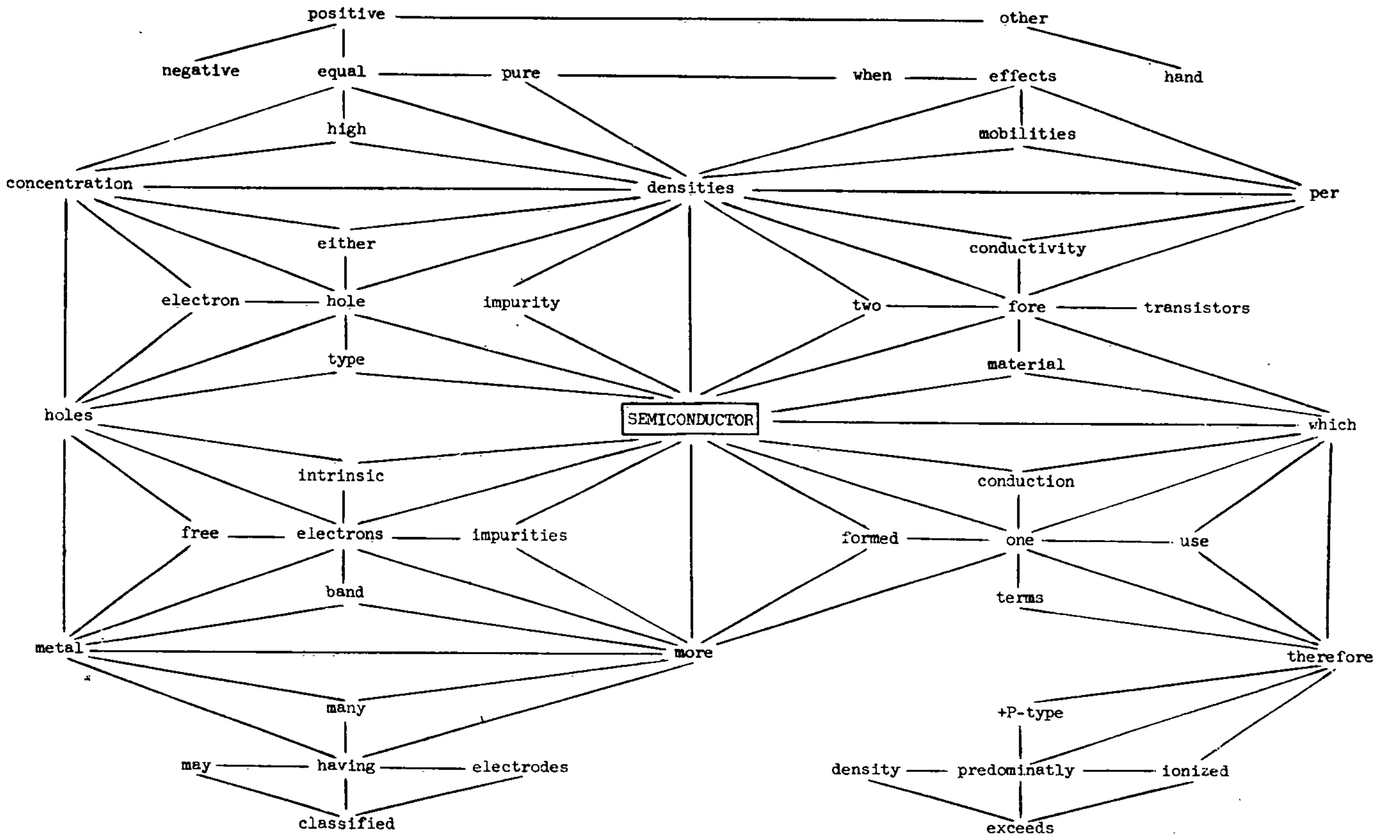


Map No. 10. General Map of Articles on Semiconductors Formula: $C_{ab} = \frac{F}{A + B - F}$



Map No. 11. General Map of Articles on Semiconductors and Transistors

Formula: $C_{ab} = \frac{F}{A + B - F}$



Map No. 12. Transistors and Semiconductors Map Based on Keyword "Semiconductor"

Formula: $C_{ab} = \frac{F}{A + B - F}$

UNCLASSIFIED

System Development Corporation,
Santa Monica, California
COMPUTER GENERATION OF WORD
ASSOCIATION MAPS FOR MAN-MACHINE
COMMUNICATION.
Scientific rept., SP-1153, by
C. Watson. 25 March 1963, 24p.,
12 maps

Unclassified report

DESCRIPTORS: Information Retrieval.
Computers.

Considers the problems of generating and displaying association maps through the use of a digital computer.

UNCLASSIFIED

UNCLASSIFIED

States that the term "association maps" denotes a two-dimensional array of elements with lines drawn between the elements to indicate an associative relationship. Reports that the association map is used for man-machine communication in an information retrieval system.

UNCLASSIFIED