IV. Correlation Measures

K. Reitsma and J. Sagalyn

Abstract

In this study the performance of ten matching functions is investigated. The performance is measured in terms of recall and precision. All ten functions are tested on the 82 document ADI collection; the best four are tested again on the larger 200 document Cranfield collection. It is shown that the Parker-Rhodes-Needham function has the best performance in the ADI collection below 0.50 recall; however, this function is the worst in the Cranfield collection test. Overall, the Cosine function shows the best performance.

1. Introduction

A document retrieval system, from a user's point of view, takes a request for information, in the form of a short verbal description, matches the request against the documents in the collection and returns those which by some measure are most relevant.

Within the SMART system, all the documents have been analyzed automatically according to word frequency counts of keywords contained in a thesaurus. Each analyzed document is represented by a description vector of concept numbers with corresponding weights (the weight being proportional to the frequency of occurence of that concept). When a request is received, it

too is analyzed in the same manner as the documents and is represented by a description vector of concept numbers and corresponding weights.

Within the system, request-document comparisons are made using a mathematical correlation coefficient. Each document is compared with the request by calculating the magnitude of the coefficient. The documents are then ranked according to the coefficient and hopefully according to the degree of relevancy with the request.

The subject of this study is to evaluate several correlation coefficients to determine which one is the "best" to use. The "best" coefficient should be the one for which the largest number of relevant documents are found on the top of the ranked document list.

Some work has been done with various correlation coefficients. In 1966, Manning and Hall analyzed several correlation coefficients and proposed two of their own, however, they did not present any conclusive evidence for an evaluation. It is the aim of this study to evaluate several of the coefficients previously used by Manning and Hall, including one which they proposed, as well as a few others which have been derived from other types of coefficients.

The initial evaluation has been done on the ADI collection of 82 documents and 35 requests. Since this collection is small, any conclusion must be verified on a larger collection, such as the 200 document Cranfield collection.

2. Weighted versus Logical Description Vectors

A document description vector can take on two forms. One is a logical or binary vector in which every element is either 0 or 1 . Each position in the vector represents a concept (e.g. the first position represents

concept number one, the third position concept three, etc.). Thus for binary vectors, if a 0 occurs in the third position, concept three is absent from that document, and if a 1 occurs in the third position, concept three is present.

The second type of vector is a weighted vector. Ideally, the positions in the vector have the same interpretation as for binary vectors. The difference is that the value in each position is 0 if the concept is not found in the document, or some integer j where $j > 0$ is proportional to the number of times the concept appears in the document. In the SMART system a weight of 12 is given to concept k if concept k occurs once in the document, 24 if k occurs twice, etc. Since approximately 600 concepts occur in the thesaurus, each document description vector would normally have a length of 600 positions. To reduce the memory space needed, only those concepts with non-zero weights are retained in the vector, the concept number and weight both being packed into the same memory location.

The use of a correlation coefficient in the two systems poses some problems. A coefficient defined for binary vectors may have a specific interpretation, either logical or statistical. However, the same coefficient used with weighted vectors may lose its former interpretation. For example, given the two vectors

$$\underline{v} = (1,1,0,1,0,1,0)$$
$$\underline{w} = (1,0,1,1,0,1,1)$$

the expression

$$\sum_i \underline{v}_i \underline{w}_i \qquad (1)$$

is interpreted as the number of matching terms or the number of concepts

the two vectors have in common. The summation equals 3, meaning there are three concepts found both in $\underline{v}$ and $\underline{w}$, namely concepts 1, 4, and 6.

However, the same expression used with weighted vectors does not produce the same simple interpretation. For example, given the two vectors

$$\underline{v} = (12,24,0,36,0,12,0)$$
$$\underline{w} = (24,0,12,24,0,12,36)$$

the above equation (EQ-1) gives a value of 1296. Although each of these vectors contains the same concepts as the binary vectors above, and each have the same three concepts in common, there is no simple interpretation for the number 1296. The closest interpretation is that it produces a relative value which can be compared with another figure derived by using the summation on $\underline{v}$ and some other vector $\underline{w}'$ as a measure of the matching concepts, thereby it determines which vector, $\underline{w}$ or $\underline{w}'$, matches better with $\underline{v}$.

An example of an expression which doesn't lose its meaning when weighted vectors are used instead of binary vectors is the following

$$\left(\sum_{i=1}^{t} \underline{v}_i^{\,2} \right)^{\frac{1}{2}} \qquad (?)$$

This expression represents the absolute length of the vector in t-space, where t is the number of concepts possible in the description vector.

There exist coefficients other than these two to measure the similarity between documents. For the most part, these coefficients are used in thesaurus construction and measure the similarity between concepts. When calculating the term - term association coefficient, several of the expressions discussed above have a different interpretation. For example, given the term description vectors $\underline{c}_1, \underline{c}_2, \ldots$ where for each term vector

the elements are denoted by a second subscript, i.e. for term i , the vector $\underline{c}_i = (\underline{c}_{i1}, \underline{c}_{i2}, \cdots, \underline{c}_{ik})$.

$$\sum_k \underline{c}_{ik}\underline{c}_{jk} \qquad \text{for all } i,j \qquad (3)$$

$$\sum_k \underline{c}_{ki}\underline{c}_{kj} \qquad \text{for all } i,j \qquad (4)$$

The first summation gives the number of documents having both terms i and j . The second summation gives the number of terms that documents i and j have in common. It is identical to expression (1) discussed previously.

3. The Correlation Coefficients

This section contains an analysis of the various correlation coefficients considered in this study. Each is analyzed according to its origin, initial interpretation, modifications made and final interpretation as a document - document correlation coefficient.

It must be noted that there is a basic difference between the document description vector and the request description vector. The former is taken from an abstract of the article which may consist of several sentences. The latter is taken from a very short request. In the 82 document ADI collection, the maximum number of concepts in one description vector is 44, the maximum weight found in 96. Among the 35 requests the maximum number of concepts in one description vector is 11, the maximum weight found is 48. Actually, most of the weights in the request vectors are 12. It is seen therefore that the document description space is not the same as the request description space. This must be kept in mind when analyzing the

various coefficients.

A) The Inner Product

Perhaps the simplest matching function is the inner product. It is defined for two vectors $\underline{v}$ and $\underline{w}$ as

$$\underline{v} \cdot \underline{w} = \sum_i \underline{v}_i \underline{w}_i$$

This is the same expression mentioned previously and denoted as (1) and (4).

If $\underline{v}$ and $\underline{w}$ are binary vectors, then the inner product is equal to the number of terms both vectors have in common. When weighted vectors are used, much of the significance of the inner product is lost. It no longer is a measure of the number of concepts found in both $\underline{v}$ and $\underline{w}$. It does, however, give a relative measure of the total weight of the matching concepts, although it poses some problems since it is not normalized. For example if

$$\begin{aligned} \underline{v} &= (\ 0,\ 2,12) \\ \underline{w}_1 &= (\ 0,13,\ 1) \\ \underline{w}_2 &= (\ 0,\ 1,\ 3)\ , \end{aligned}$$

the inner products are

$$\begin{aligned} \underline{v} \cdot \underline{w}_1 &= 2(13) + 12(1) = 38 \\ \underline{v} \cdot \underline{w}_2 &= 2(\ 1) + 12(3) = 38\ . \end{aligned}$$

The inner product of $\underline{v}$ with both $\underline{w}_1$ and $\underline{w}_2$ gives 38, even though the two w-vectors are very different. Because of these problems, the inner product is not used in the evaluation; it is mentioned here since it forms the basis of several other coefficients.

B) The Cosine Coefficient

This function was proposed by Salton and has the following form

$$C = \frac{\sum_{i=1}^{t} \underline{v}_i \underline{w}_i}{\left[\sum_{i=1}^{t} (\underline{v}_i)^2 \cdot \sum_{i=1}^{t} (\underline{w}_i)^2 \right]^{\frac{1}{2}}}$$

It is used as a term - term association coefficient as well as a document - document correlation coefficient. In both cases its interpretation is the same. If $\underline{v}$ and $\underline{w}$ are t-dimensional vectors, then C is the direction cosine in the term space or document space of the angle subtended by the vectors $\underline{v}$ and $\underline{w}$. The interpretation also does not depend on the type of vectors used, whether they be binary or weighted.

Since the denominator is the product of the absolute lengths of the vectors in t-space, it increases with an increase in the vector length. If the two vectors are increased in length, the inner product will increase by an amount equal to or less than the denominator. Since the possible number of matching concepts tends to increase with increased vector length and since the cosine correlation generally decreases, this function has at least one serious fault, i.e. length dependency.

C) The Hypersine Coefficient

This function was proposed in the work of Hall and Manning and is designed to reduce the length dependency of the cosine function. The Hypersine function is

$$HS = \frac{\sum_{i=1}^{t} \underline{v}_i \underline{w}_i \underline{w}_i}{\left[\sum_{i=1}^{t} \underline{v}_i \underline{v}_i \underline{w}_i \cdot \sum_{i=1}^{t} \underline{w}_i \underline{w}_i \underline{w}_i \right]^{\frac{1}{2}}}$$

This correlation coefficient is basically the same as the Cosine coefficient already described except that within each summation, another factor has been added. The numerical effect of this added $\underline{w}_i$ factor is zero since it can be divided out. The effect of the factor is to reduce the magnitude of the document vector length (the vector $\underline{v}$ being the document description vector), since the product $\underline{v}_i \underline{v}_i \underline{w}_i$ is zero when $\underline{v}_i > 0$ but $\underline{w}_i = 0$. This term therefore is positive only when both $\underline{v}_i$ and $\underline{w}_i$ are greater than 0, i.e. when concept i is found in both $\underline{w}$ and $\underline{v}$. In other words, the length of the document vector is calculated in the subspace of the request space. Since the document vector is usually much longer than the request vector, the Hypersine reduces the dependency on length of the Cosine function.

D) The Overlap Coefficient

In an effort to measure the amount of overlap between two vectors, the following formula was proposed

$$OL = \frac{\sum \min (\underline{v}_i , \underline{w}_i)}{\min \left(\sum \underline{v}_i , \sum \underline{w}_i \right)}$$

It was originally proposed for binary term vectors, where the summations are taken over $i=1,\ldots,t$, where t equals the number of documents in the collection.

Without any modifications, the function may be used with weighted vectors, in which case the summations are taken over $i=1,\ldots,d$, where d equals the number of concepts in the description vector.

The numerator of the function is the smallest vector in the document space consisting of elements from $\underline{v}$ and $\underline{w}$. It is divided by the smallest vector, either $\underline{v}$ or $\underline{w}$. In the case of weighted vectors, "smallest" means the least sum of weights.

E) The Maron-Kuhns Coefficient

This formula was originally proposed as a measure of association between index terms. Used with binary vectors, it measures the number of matching terms for two given term description vectors over and above the number of matching terms expected for purely random vectors. The formula is

$$\text{M-K} = \frac{\sum \underline{v}_i \underline{w}_i \cdot \sum \bar{\underline{v}}_i \bar{\underline{w}}_i - \sum \underline{v}_i \bar{\underline{w}}_i \cdot \sum \bar{\underline{v}}_i \underline{w}_i}{\sum \underline{v}_i \underline{w}_i \cdot \sum \bar{\underline{v}}_i \bar{\underline{w}}_i + \sum \underline{v}_i \bar{\underline{w}}_i \cdot \sum \bar{\underline{v}}_i \underline{w}_i}$$

where the symbol $\bar{\underline{w}}_i$ is the complement of $\underline{w}_i$, that is, if $\underline{w}_i = 0$, $\bar{\underline{w}}_i = 1$ and if $\underline{w}_i = 1$, $\bar{\underline{w}}_i = 0$. All summations are taken from $i = 1$ to t, where t equals the number of documents. The vectors $\underline{v}$ and $\underline{w}$ are binary term description vectors.

The numerator can be written as $t\delta$ where

$$\delta = \sum v_i w_i - \frac{\sum v_i \sum w_i}{t}$$

by making the simple substitution $\sum \underline{v}_i = \sum \underline{v}_i \underline{w}_i + \sum \underline{v}_i \overline{\underline{w}}_i$ and a similar substitution for $\sum \underline{w}_i$. The first term in the expression for δ gives the number of documents containing both terms $\underline{v}$ and $\underline{w}$ and the second term is proportional to the frequency of documents both having terms $\underline{v}$ and $\underline{w}$ if both $\underline{v}$ and $\underline{w}$ were random vectors.

For random vectors $\delta = 0$ giving a value of 0 for the coefficient. For vectors in which there are a greater or smaller number of matching documents the expected number δ is greater than or less than 0 . The range of the function is then $-1 \leq M\text{-}K \leq +1$, $+1$ signifying perfectly correlated terms and -1 signifying perfectly uncorrelated terms.

When the Maron-Kuhns coefficient is modified to be used as a document - document correlation coefficient, its interpretation is altered. The summations must now be taken from $i = 1$ to d where d equals the number of concepts in the description vector. The formula then gives a measure of the number of concepts found in both document $\underline{v}$ and document $\underline{w}$ over and above the number expected if both $\underline{v}$ and $\underline{w}$ were random vectors.

Further problems arise when the document description vectors are weighted vectors instead of binary vectors. One problem is the question of complementation. To solve this problem, the complement of an element of a vector, is defined as the maximum concept weight found in the entire collection in which that vector is found minus the concept weight to be complemented.

A second problem is concerned with all the zero elements of the document description vector. If the above method of complementation were used, the complement of a concept weight of zero would equal the maximum

concept weight in the collection. If the number of concepts in the thesaurus is large and the number of concepts in any document description vector is much smaller, a large number of zero elements will occur in a vector. When these elements are complemented, all the elements will equal the maximum concept number. In this case, the summation $\sum \bar{v}_i \bar{w}_i$ will be very large and its product with $\sum v_i w_i$ will be much larger than $\sum v_i w_i \cdot \sum \bar{v}_i \bar{w}_i$, giving a coefficient which will always be near 1. To avoid this problem, only non-zero concepts are complemented.

In the ADI collection the maximum document weight is 96 and the maximum query weight is 48. The complement for an element in a document vector or a query vector is respectively

$$\bar{v}_i = 96 - v_i$$
$$\bar{w}_i = 48 - w_i$$

if v_i or w_i is greater than zero, otherwise the complement is zero.

One further alteration made, in order to avoid negative correlation coefficients, results in a change in the range of the formula. It has been adjusted so that the range is from 0 to +1 by adding 1 to the unadjusted coefficient and dividing by 2 .

F) The Parker-Rhodes-Needham Coefficient

This formula was originally proposed as an index term - index term association measure for use with binary term vectors. The function is

$$\text{P-R-N} = \frac{\sum \underline{v}_i \underline{w}_i}{\sum \underline{v}_i + \sum \underline{w}_i - \sum \underline{v}_i \underline{w}_i} ,$$

where all summations are taken over $i = 1$ to t, and where t equals the number of documents in the collection. Since the term vectors are binary, the interpretation of the terms in the denominator is simple. The first term is the number of documents containing term $\underline{v}$, the second is the number of documents containing term $\underline{w}$, and the third is the number of documents containing both terms $\underline{v}$ and $\underline{w}$. On the whole, the denominator gives the number of documents containing at least one of the terms.

For two identical terms, the denominator equals the numerator and the association is 1. For two independent terms, where a document does not contain both terms, the numerator is zero and the association is 0.

When term - term associations are calculated, all the terms are usually compared with all the other terms at the same time, using matrix multiplication. The result is a matrix whose elements are terms of the above formula. Since matrix multiplication requires the calculation of many inner products, each of the entries in the association matrix is the result of an inner product and therefore, so is each term in the P-R-N formula. Thus, the summations $\sum \underline{v}_i$ and $\sum \underline{w}_i$ are in practice calculated by $\underline{v} \cdot \underline{v}$ and $\underline{w} \cdot \underline{w}$ which is the same as $\sum \underline{v}_i \underline{v}_i$ and $\sum \underline{w}_i \underline{w}_i$ which is the same as $\sum (\underline{v}_i)^2$ and $\sum (\underline{w}_i)^2$, where the summations are taken

as before.

The above discussion is necessary when describing the modifications made to use the P-R-N formula for document - document correlations. With weighted, document descriptions vectors the formula becomes

$$\text{P-R-N} = \frac{\sum \underline{v}_i \underline{w}_i}{\sum (\underline{v}_i)^2 + \sum (\underline{w}_i)^2 - \sum \underline{v}_i \underline{w}_i}$$

where the summations are taken from $i = 1$ to d, and where d equals the number of concepts in the description vector.

The interpretation of this function is not simple. The closest meaning which can be attached to the denominator is that it represents twice the maximum weight of the inner product of the two vectors, assuming perfect correlation, minus the actual inner product. The difference, therefore, will always be greater than or equal to the actual inner product.

By the argument previously used to determine the range of the binary P-R-N function, the range of this function can be shown to be from 0 to 1, inclusive.

G) The Stiles Coefficient

The Stiles function incorporates the parameter δ as it was defined for the Maron-Kuhns function. The formula is

$$\text{St} = \text{Log}_{10} \left\{ \frac{n(\, |n\delta| - \frac{n}{2})^2}{\left(\sum \underline{v}_i \right) \left(\sum \underline{w}_i \right) \left(n - \sum \underline{v}_i \right) \left(n - \sum \underline{w}_i \right)} \right\} .$$

Since the formula was originally proposed to calculate index term - index

term associations, n equals the number of documents in the collection and all the summations are taken from i = 1 to n . Stiles defines his formula as based upon the chi-square formula and gives the distance from the expected frequency of occurrence assuming no association. The magnitude of this function may be greater than 1 due to the presence of the Log function.

By a simple analysis, it can be seen that the four factors in the denominator are the number of documents containing term v , and the number containing term w , the number not containing term v , and the number not containing term w , respectively.

This formula has been adapted for use with weighted vectors. The modified formula is

$$St = \ln_e \left\{ \frac{N\left[\left| N\left(\frac{\sum v_i w_i}{144} \quad \frac{\sum v_i^2 \quad w_i^2}{144 \cdot N} \right) \right| - \frac{N}{2} \right]^2}{\frac{\sum (v_i)^2}{144} \cdot \frac{\sum (w_i)^2}{144} \cdot \left[N - \frac{\sum (v_i)^2}{144} \right] \cdot \left[N - \frac{\sum (w_i)^2}{144} \right]} \right\} .$$

Ignoring the factor of 144 , the function is the same as Stile's original function except that the denominator contains the sum of squares instead of only the sum of the terms. The reason for this change has already been explained in the discussion of the Parker-Rhodes-Needham coefficient. One other variation from the original function is the use of the natural logarithm instead of the base 10 log. This substitution was made in order to facilitate coding on the computer, where a natural logarithm function exists. No difficulty should arise since both logarithms are increasing functions.

The use of the factor 144 is intended to simulate the original function. In essence, dividing by 144 partially eliminates the effect of weights and therefore approximates binary terms.

The definition of N presents some problems. Originally, it was intended to let N equal the number of concepts in the thesaurus, about 610. However, if this were done, it is possible that the last two factors in the denominator might become negative. Therefore, to avoid this problem. N is defined as (4)(610), the (4) being the average concept weight divided by 12, the base of the weighting system. (48 was arbitrarily chosen as the average concept weight.) The coefficient is assured of being real, and no attempt to normalize it has been made, so that values greater than 1 are possible.

H) The Average Coefficient

This formula simply calculates the average weight of all those concepts which are found in both description vectors $\underline{v}$ and $\underline{w}$. The formula is

$$Av = \frac{\sum (\underline{v}_i + \underline{w}_i)(\delta_i)}{2 \cdot N}$$

where

$$\delta_i = \begin{cases} 1 & \text{if both } \underline{v}_i \text{ and } \underline{w}_i > 0 \\ 0 & \text{if either } \underline{v}_i \text{ or } \underline{w}_i = 0 , \end{cases}$$

and where N equals the number of matching concepts. The summation is taken from $i = 1,\ldots,d$, where d equals the number of concepts in the description vector.

It was originally intended to use this function, time permitting, to determine whether it is more important to have fewer matching concepts at higher weights than it is to have more matching concepts at lower weights.

I) The Reitsma-Sagalyn Coefficient

This function is based on the idea that the relative weights of the matching concepts are very important. In other words, it is more important to have weights of matching concepts equal rather than unequal. The function is

$$R\text{-}S = \frac{\sum_{i=1}^{t} \frac{\min(v_i, w_i)}{\max(v_i, w_i)}}{N} ,$$

where N equals the number of matching concepts. As an alternative, N may equal the maximum number of concepts found in the document or request description vector.

The range of this function is 0 to 1, where 0 indicates no correlation and 1 indicates perfect correlation.

This function can be used with either binary or weighted vectors.

The main problem with this function is that it depends entirely upon the relative weights of the matching concepts. As described in the beginning of this section, the requests are usually much shorter than the abstract from which the document descriptions are taken. It follows, then, that the relative weights of the concepts in the request vector do not indicate the relative importance of the concepts, i.e. many weights tend to be the same and, therefore, the relative importance of the various concepts cannot be determined.

It therefore seems reasonable that if this coefficient were used in a system with a relevance feedback system, it might prove more powerful. Ideally, it should be used in a system where the requests are such that the user indicates the relative importance of various keywords in his request,

and all the request weights are adjusted so that the request space approximately equals document space. (i.e. adjusting the request weights so that the average weight among the requests equals the average weight among the documents.)

4. Method of Evaluation

The power of the various correlation coefficients is determined by the use of recall - precision plots. Recall is defined as the proportion of relevant documents retrieved, while precision is defined as the proportion of retrieved documents which are actually relevant.

$$\text{Recall} = \frac{\text{number of documents retrieved and relevant}}{\text{total number of relevant documents}}$$

$$\text{Precision} = \frac{\text{number of documents retrieved and relevant}}{\text{total number of documents retrieved}}$$

For each of the queries, a recall - precision graph is produced. These are then averaged over all the queries. The method of averaging is as follows:

1) the peaks of each recall - precision graph are connected and the first peak is extrapolated horizontally to the y-axis (precision axis where recall equals zero);

2) the value of precision along this constructed line is thus determined at twenty different points along the recall axis, **i.e.** at recall equal to .05, .10, .15, ..., .95, 1.00;

3) for each of these points, the precision is averaged over all the queries;

4) a final graph is plotted along these twenty average precision values.

An averaged recall - precision graph is obtained in the above

for every coefficient to be evaluated. The final evaluation is based on the comparison of these average graphs. The coefficient which produces an average recall-precision graph above and to the right of all the other graphs is assumed to be the best coefficient, with respect to that document collection and that set of queries, since for any value of recall, the precision is higher than for any other coefficient and for any value of precision, the recall is higher than for any other coefficient.

It is possible that two average recall-precision graphs may coincide or intersect. In the former case, no conclusion can be made as to which coefficient is better. In the latter case, one coefficient may be better than another for a given range of recall. For example, is the coefficient D_1 gives an average recall-precision graph above that for coefficient D_2 in the range of recall from 0 to .40 , it may be concluded that coefficient D_1 is better than D_2 when the user is interested in the first 40% or less relevant documents. If, however, the user is interested in finding 50%, 75%, or 100% of the relevant documents, D_1 is no longer the most powerful coefficient. The best performance, in this, might result from the use of D_1 to find the first 40% relevant documents and then the use of D_2 to find the remaining relevant documents.

One danger exists in using two different coefficients to process a request. It may happen that the specific documents contained in the first 40% retrieved by D_1 are the same documents which D_2 retrieves last. In that case, using the two together might not give the desired performance. Most probably, coefficient D_1 would be used if the user were only interested in 40% or less of the relevant documents. If he were interested in more, D_2 might be used.

5. Experimental Results

The following functions have been tested with the use of the ADI collection, making four comparisons in each test as follows:

Table 1 – Overlap, Cosine, Parker-Rhodes-Needham, Reitsma-Sagalyn.

Table 2 – Average, Stiles, Reitsma-Sagalyn (sorted up), Cosine.

Table 3 – Cosine, Hypersine, Maron-Kuhns, Reitsma-Sagalyn (modified).

with the Cranfield collection:

Table 4 – Overlap, Cosine, Parker-Rhodes-Needham, Stiles.

The tables contain averages, from which the average recall - precision graphs were made, and the standard deviation (S.D.D.) of the averages.

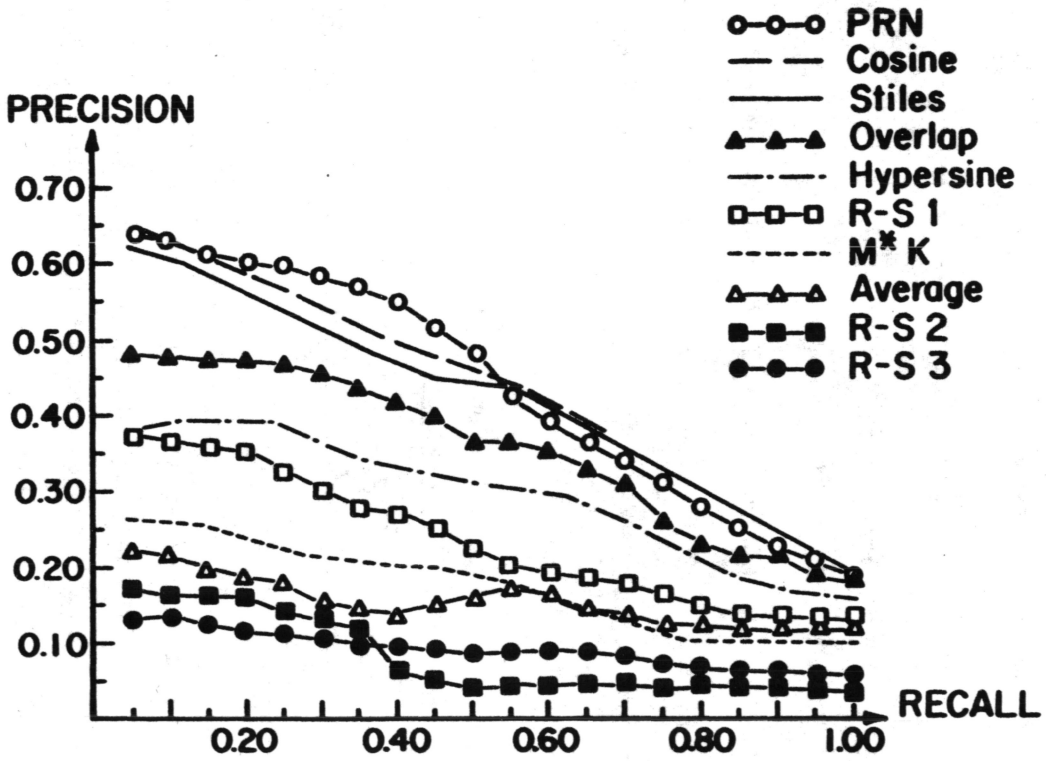
The data in the tables are summarized in Figure 1 which shows the performance of all coefficients tested on the ADI collection.

Recalling the discussion of the recall and precision measures as a means for evaluating the performance of different correlation coefficients, Figure 1 shows the following output:

1) Three correlation functions exhibit a decidedly better performance than the others. They have been replotted on a larger scale on Figure 3 to show the difference in behavior in more detail. The functions are Stiles, Cosine and Parker-Rhodes-Needham.

2) In the recall interval 0 - 0.50 the Parker-Rhodes-Needham coefficient has a better performance than the other two; in the recall interval above 0.50 the performance of this function is worse than the others. This indicates that the Parker-Rhodes-Needham function gives the best results in a system with a cutoff value smaller than 0.50 .

3) Comparing the Cosine and Stiles coefficients, the former has a better performance below 0.55 recall, while at higher recall values, the performance of both functions is almost identical. Therefore, in the entire interval, the Cosine coefficient is better than the Stiles function.



Performance of all the correlation functions, using the ADI Collection.

Fig. 1

Explanation of the Terms:

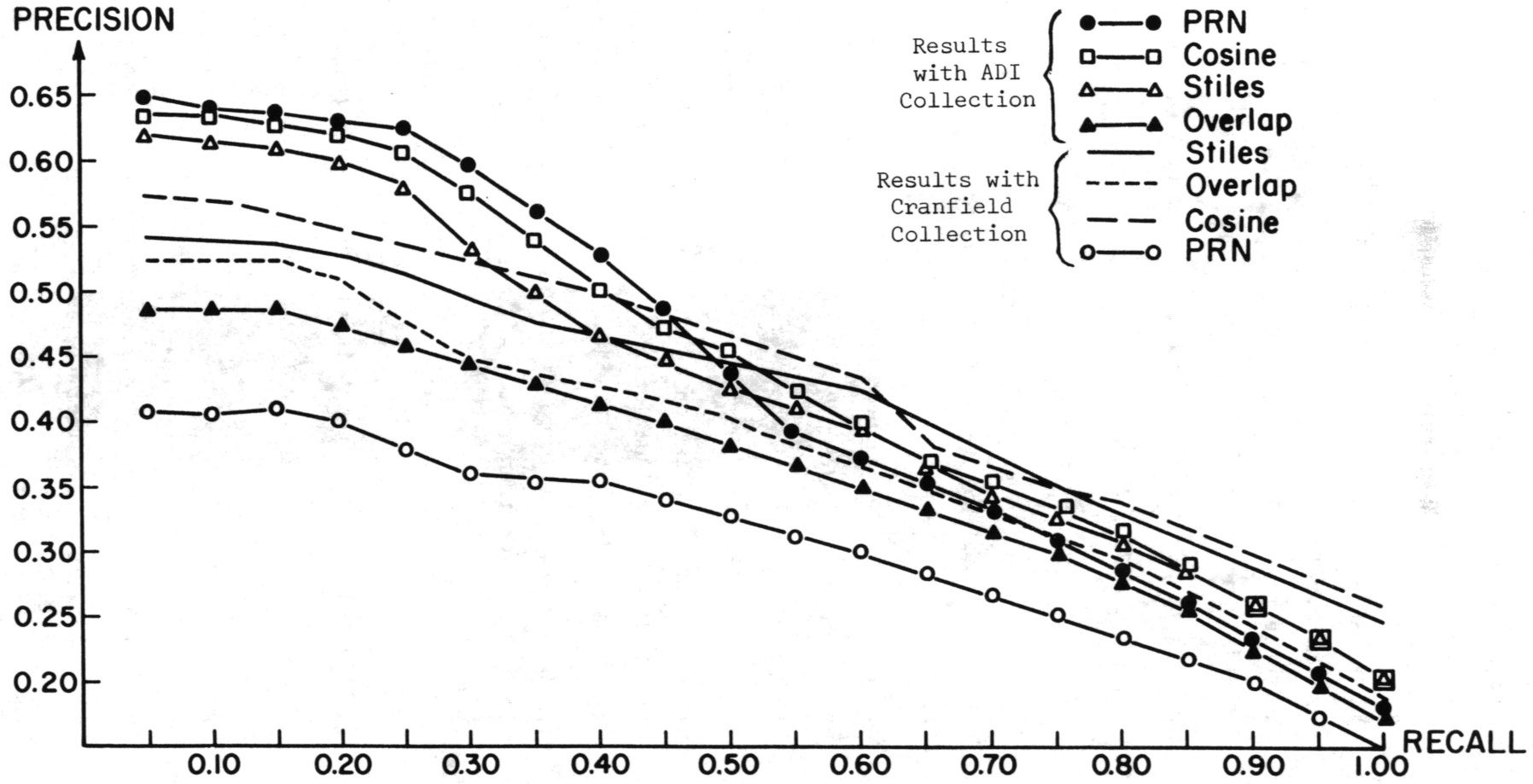
PRN Parker-Rhodes-Needham

M*K Maron and Kuhns

R-S 1 Own, div. by max # of concepts in document or query

R-S 2 Own, sorted up

R-S 3 Own, div. by # of matching concepts



Comparison of the results obtained with the four best correlation functions using the ADI and CRANFIELD collections.

Fig. 3

4) These conclusions are further supported by the almost equivalent values of the standard deviations of the respective functions.

5) The other functions show a performance strictly below the above mentioned coefficients. Only the Overlap coefficient approaches the three best and only above 0.75 recall which region is fairly insignificant in practice.

However, the four best functions, when tested with the Cranfield Collection, exhibit a different behavior:

1) The differences between the functions have increased.

2) The Cosine function shows a better performance than the other three (i.e. the Parker-Rhodes-Needham, Stiles, and Overlap coefficients).

3) The Parker-Rhodes-Needham is not close to the Cosine anymore; it is the worst of the four.

4) The performance of the Overlap is no longer the worst, in fact, it remains very close to the Cosine and Stiles coefficient.

5) The standard deviation of the Cosine function is much smaller than for the other functions. This supports the conclusion that this function is better than the rest in this collection.

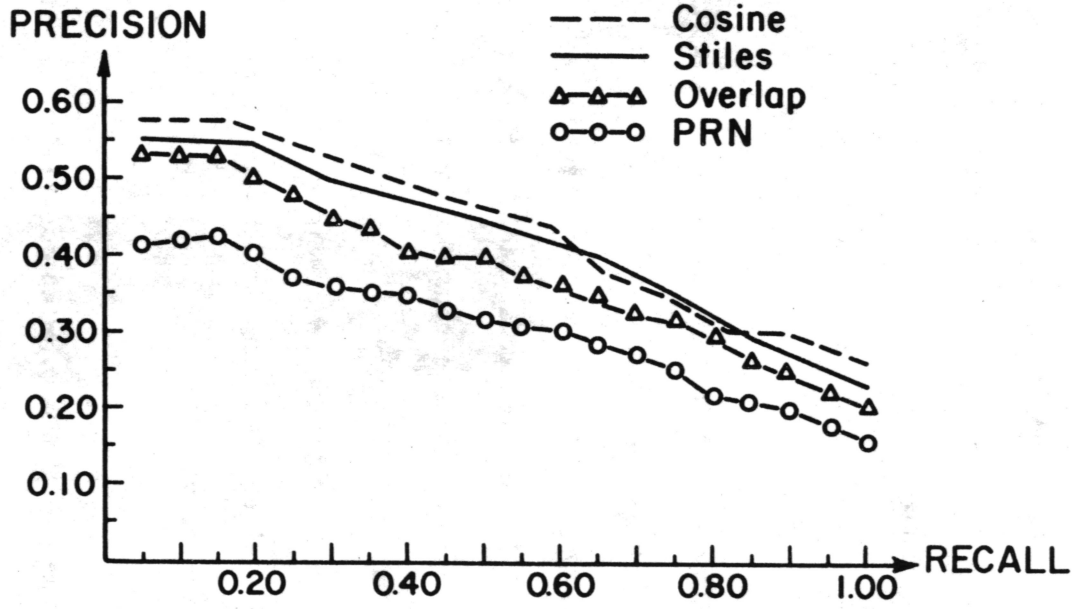
6) The overall precision at the same recall is lower in the Cranfield collection than in the ADI collection.

6. Discussion

In this section, an attempt is made to explain the behavior of the various coefficients and to suggest possible modifications for future investigations.

Cosine:

The Cosine function shows a consistently high performance in both the ADI and Cranfield collections. Since it is length dependent and since the Hypersine tries to reduce this dependence unsuccessfully, a compromise some-



The performance of the four best correlation functions with the CRANFIELD collection.

Fig. 2

where between the Cosine and Hypersine may prove effective is length dependency inhibits the efficiency of the Cosine function.

Hypersine:

It is seen that the performance of the Hypersine function is worse than the Cosine. Therefore, it seems as though the non-matching concepts of the document which were deleted in calculating the document vector length are indeed important. Evidently, some degree of length dependency is beneficial in a matching function and the Hypersine tries to eliminate this dependency incorrectly and to too great a degree.

Maron-Kuhns:

The performance of this function is far below the three good functions. There are two possible explanations. One is the problem of complementation, i.e. the complement of a weighted vector may be defined in a better way. The second possible explanation is the importance of the non-zero non-matching weights. In this study, only the matching weights were complemented. It might be advisable to complement the zero weights in one vector for those concepts with non-zero weights in the other vector. It still does not seem advisable to complement all the zero weights for the same reasons as stated previously.

Overlap:

The performance of the Overlap coefficient in the ADI and Cranfield varies drastically. The explanation may lie in the differences between the subject content of the two collections. Since the weights of the request are usually less than the weights of the document, the numerator is not strongly influenced by a matching concept with a very large weight in a

document. This insensitivity may explain the poor performance of this function compared to some of the other coefficients.

Parker-Rhodes-Needham:

The striking difference in performance of this function in the ADI collection, where it proved very powerful, and in the Cranfield collection where it performed rather poorly is puzzling. Further evaluation with other document collections is needed before any conclusions as to its value can be made.

Stiles:

This coefficient shows a consistent high performance for both the ADI and Cranfield collections. It is far less sensitive to variations in collection characteristics than the Overlap and the Parker-Rhodes-Needham coefficients. The explanation of this phenomenon is difficult due to the complexity of the formula; however, its quasi-binary character seems to give reasonable results. One possible refinement may be a better definition of N.

Reitsma-Sagalyn:

Three different modifications of this formula were used in this study. In one of them N equals the number of concepts in either the query vector or the document vector (the maximum of the two). Another form results in using the number of matching concepts for N. When this is done, it is observed that many relevant documents occur at the end of the ranked list. This leads to the third modification in which the second form was used but the documents were ranked in the reverse order. In general, this formula proved ineffective.

References

[1] Vincent E. Giuliano, Automatic Message Retrieval by Associative Techniques, Working Memorandum ACORN-17, Arthur D. Little, Inc., Cambridge, Mass., November 13, 1962.

[2] James A. Manning and George A. Hall, An Analysis of Matching Functions, Computer Science 435, Cornell University, June 2, 1966.

[3] M. E. Maron and J. L. Kuhns, "On Relevance, Probalistic Indexing and Information Retrieval", Journal of the ACM, Vol. 7, 1960, pp. 216-244.

[4] Joseph Spiegel and Edward Bennett, "A Modified Statistical Association Procedure for Automatic Document Content Analysis and Retrieval", Statistical Association Methods for Mechanized Documentation, Symposium Proceedings 1964, National Bureau of Standards, Washington, D.C., December 15, 1965, pp. 47-60.

[5] J. Spiegel, E. Bennett and E. Haines, Statistical Association Procedures for Message Content Analysis, The Mitre Corporation, Bedford, Mass., April 1963.

[6] H. E. Stiles, "The Association Factor in Information Retrieval," Journal of the Association for Computer Machinery, Vol. 8, 1961, pp. 271-279.

The 82 Document ADI Collection

Averages Over 35 Requests, Abstracts Thesaurus-1

	Overlap	Cosine	P-R-N	R-S 1
Normalized Recall	.7781	.8006	.7974	.5355
Normalized Precision	.5651	.6086	.6073	.2666

RECALL	OVERLAP		COSINE		P-R-N		R-S 1	
	PREC	S.D.	PREC	S.D.D.	PREC	S.D.D.	PREC	S.D.D.
.05	.4845	.3761	.6400	.3404	.6435	.4229	.1288	.3880
.10	.4841	.3756	.6399	.3406	.6355	.4242	.1308	.3889
.15	.4865	.3768	.6260	.3479	.6286	.4295	.1231	.3795
.20	.4843	.3756	.6097	.3524	.6188	.4396	.1190	.3738
.25	.4726	.3685	.5840	.3437	.6039	.4354	.1170	.3626
.30	.4573	.3595	.5527	.3221	.5843	.4093	.1119	.3549
.35	.4400	.3534	.5214	.3092	.5549	.3830	.1070	.3497
.40	.4142	.3410	.4915	.2917	.5118	.3398	.1041	.3435
.45	.3953	.3343	.4663	.2758	.4761	.3084	.0967	.3386
.50	.3870	.3329	.4448	.2687	.4452	.2935	.0913	.3325
.55	.3743	.3195	.4279	.2546	.4208	.2649	.0944	.3166
.60	.3556	.3066	.4097	.2449	.3929	.2438	.0949	.3032
.65	.3380	.2986	.3898	.2417	.3669	.2307	.0957	.2929
.70	.3192	.2862	.3617	.2330	.3417	.2208	.0944	.2792
.75	.3018	.2736	.3319	.2309	.3159	.2173	.0916	.2625
.80	.2805	.2547	.3103	.2038	.2934	.1861	.0893	.2412
.85	.2593	.2385	.2867	.1742	.2716	.1581	.0876	.2244
.90	.2358	.2223	.2616	.1487	.2482	.1350	.0857	.2091
.95	.2104	.2027	.2340	.1310	.2223	.1210	.0846	.1900
1.00	.1857	.1883	.2070	.1244	.1969	.1195	.0832	.1750

Recall-Precision Table
(ADI Collection)

Table 1

The 82 Document ADI Collection

Averages Over 35 Requests, Abstracts Thesaurus-1

	Average	Stiles	R-S 2	Cosine
Normalized Recall	.6978	.7928	.4645	.8006
Normalized Precision	.3891	.5981	.2345	.6086

RECALL	AVERAGE		STILES		R-S 2		COSINE	
	PREC	S.D.	PREC	S.D.D.	PREC	S.D.D.	PREC	S.D.D.
.05	.2185	.2524	.6151	.3711	.1612	.3183	.6400	.3534
.10	.2090	.2469	.6133	.3744	.1564	.3208	.6399	.3580
.15	.2014	.2283	.6037	.3548	.1587	.3056	.6260	.3333
.20	.1862	.1904	.5909	.3441	.1570	.2723	.6097	.3222
.25	.1713	.1553	.5628	.3335	.1474	.2198	.5840	.3166
.30	.1580	.1341	.5283	.3082	.1326	.1770	.5527	.3002
.35	.1486	.1290	.4974	.2926	.1107	.1440	.5214	.2926
.40	.1461	.1255	.4678	.2870	.0927	.1288	.4915	.2932
.45	.1489	.1260	.4501	.2891	.0851	.1274	.4663	.2919
.50	.1508	.1275	.4364	.2974	.0798	.1290	.4448	.2960
.55	.1524	.1295	.4236	.2836	.0786	.1289	.4279	.2798
.60	.1515	.1222	.4090	.2764	.0758	.1197	.4097	.2709
.65	.1481	.1130	.3879	.2725	.0743	.1098	.3898	.2680
.70	.1456	.1077	.3595	.2570	.0747	.1020	.3617	.2540
.75	.1438	.1048	.3293	.2406	.0755	.0956	.3319	.2391
.80	.1409	.1026	.3080	.2243	.0760	.0907	.3103	.2245
.85	.1370	.1009	.2853	.2085	.0773	.0868	.2867	.2089
.90	.1317	.0972	.2608	.1953	.0782	.0823	.2616	.1963
.95	.1265	.0932	.2323	.1822	.0791	.0781	.2340	.1844
1.00	.1220	.0926	.2029	.1774	.0804	.0755	.2070	.1795

Recall Precision Table
(ADI Collection)
Table 2

The 82 Document ADI Collection

Averages Over 35 Requests, Abstracts Thesaurus

	Cosine	Hypersine	M-K	R-S 3
Normalized Recall	.8006	.7794	.5308	.6187
Normalized Precision	.6086	.5231	.3381	.4056

RECALL	COSINE		HYPERSINE		M-K		R-S 3	
	PREC	S.D.	PREC	S.D.D.	PREC	S.D.D.	PREC	S.D.D.
.05	.6400	.3936	.3845	.3458	.2596	.4785	.3879	.4842
.10	.6399	.3937	.3891	.3451	.2561	.4764	.3804	.4812
.15	.6260	.3850	.3941	.3552	.2430	.4677	.3707	.4815
.20	.6097	.3784	.3904	.3546	.2346	.4588	.3536	.4728
.25	.5840	.3714	.3724	.3356	.2323	.4596	.3357	.4555
.30	.5527	.3610	.3549	.3076	.2287	.4535	.3072	.4154
.35	.5214	.3613	.3415	.2983	.2236	.4492	.2839	.3916
.40	.4915	.3665	.3361	.2962	.2154	.4373	.2633	.3664
.45	.4663	.3647	.3318	.2943	.2029	.4102	.2484	.3502
.50	.4448	.3667	.3269	.2957	.1894	.3948	.2266	.3427
.55	.4279	.3513	.3153	.2738	.1735	.3649	.2062	.3227
.60	.4097	.3398	.2998	.2595	.1564	.3471	.1881	.3073
.65	.3898	.3306	.2826	.2535	.1409	.3343	.1715	.2973
.70	.3617	.3086	.2667	.2477	.1278	.3117	.1577	.2775
.75	.3319	.2877	.2504	.2510	.1161	.2884	.1445	.2579
.80	.3103	.2725	.2320	.2298	.1053	.2737	.1348	.2428
.85	.2867	.2570	.2135	.2104	.0970	.2563	.1267	.2282
.90	.2616	.2434	.1926	.1965	.0892	.2422	.1186	.2169
.95	.2340	.2288	.1686	.1917	.0812	.2293	.1118	.2034
1.00	.2070	.2217	.1452	.1953	.0733	.2237	.1046	.1973

Recall Precision Table
(ADI Collection)

Table 3

The 200 Document Cranfield Collection

Averages Over 42 Requests, Abstracts Thesaurus-3

	Overlap	Cosine	P-R-N	Stiles
Normalized Recall	.8404	.8772	.8654	.8760
Normalized Precision	.6205	.6815	.6098	.6727

	OVERLAP		COSINE		P-R-N		STILES	
RECALL	PREC	S.D.	PREC	S.D.D.	PREC	S.D.D.	PREC	S.D.D.
.05	.5245	.4022	.5688	.4894	.4076	.4582	.5408	.4968
.10	.5245	.4022	.5688	.4894	.4084	.4576	.5408	.4968
.15	.5229	.3995	.5655	.4871	.4100	.4530	.5375	.4945
.20	.5051	.3822	.5550	.4716	.4043	.4362	.5278	.4819
.25	.4829	.3714	.5432	.4517	.3957	.4264	.5143	.4649
.30	.4578	.3607	.5301	.4215	.3813	.4126	.5015	.4391
.35	.4438	.3512	.5193	.3955	.3660	.3928	.4908	.4160
.40	.4315	.3459	.5019	.3694	.3548	.3771	.4735	.3911
.45	.4153	.3374	.4844	.3376	.3411	.3603	.4593	.3592
.50	.3962	.3375	.4680	.3198	.3294	.3571	.4459	.3409
.55	.3803	.3242	.4492	.2960	.3121	.3429	.4346	.3212
.60	.3650	.3150	.4258	.2790	.2965	.3289	.4201	.3060
.65	.3523	.3117	.3940	.2616	.2822	.3103	.3967	.2878
.70	.3382	.3096	.3668	.2487	.2700	.2970	.3707	.2714
.75	.3225	.3067	.3458	.2438	.2585	.2900	.3466	.2631
.80	.2978	.2938	.3260	.2355	.2388	.2614	.3258	.2553
.85	.2728	.2845	.3122	.2313	.2217	.2408	.3090	.2512
.90	.2472	.2794	.2967	.2258	.2039	.2260	.2903	.2457
.95	.2217	.2798	.2808	.2246	.1872	.2223	.2691	.2443
1.00	.1966	.2862	.2649	.2287	.1706	.2319	.2476	.2487

Recall Precision Table
(Cranfield Collection)

Table 4