On the Construction of Effective Vocabularies for Information Retrieval

G. Salton and Clement T. Yu[+]

Abstract

Natural language query formulations exhibit advantages over artificial language statements since they permit the user to approach the retrieval environment without prior training and without using intermediaries. To obtain adequate retrieval output, it is however necessary to emphasize the good terms and to deemphasize the bad ones. The usefulness of the terms in a natural language vocabulary is first characterized in terms of their frequency distribution over the documents of a collection. The construction of "good" natural language vocabularies is then described, and methods are given for improving the vocabulary by transforming terms that operate poorly for retrieval purposes into better ones.

I. Natural and Artificial Retrieval Languages

Most people would agree that if procedures were available for automatically analyzing the content of natural language texts, then natural language query (and document) formulations would generally be preferred in a retrieval environment over the use of artificial languages.

The following advantages are immediately apparent:

1. a user might utilize his own natural language formulations in submitting a search request and would not need to master any of the more or less compelling artificial languages;

2. no need would arise for defining a generally acceptable canonical form to represent natural language statements;

3. trained intermediaries would not be interposed between the user and the retrieval environment, and users could approach the search and retrieval system directly;

4. delays and errors in the query handling could be reduced, or entirely avoided;

5. errors which inevitably crop up in the definition of any artificial retrieval language (such as lack of specificity of the terms, or excess of specificity) would play no role in the system;

6. feedback searches in which the user interacts with the system during the retrieval operations would be easier to implement.

The use of natural language query or document formulations for information retrieval is however predicated on the availability of language processing methods which can extract adequate content indicators from natural language statements. Furthermore these methods should be applicable to substantial bodies of data, since restrictions to limited environments (such as geometric constructs, baseball scores, or airline timetables) are not realistic.

Fortunately, the content analysis problem is not as impenetrable in a retrieval environment, as it would be, for example, in language translation, since the main task consists in recognizing the subject matter, while bypassing the more complicated text evaluation problems designed to determine the truth, or falsity, or value of the various statements. In particular, some of the hardest semantic problems arising in the analysis of natural languages, including an exhaustive recognition of synonyms, and a complete disambiguation of terms and phrases might be dispensed with in most circumstances.

An approach to the construction of effective natural language vocabularies is outlined in the next few paragraphs.

II. The Specification of Term Importance

All systems for automatic natural language analysis are based on an initial selection of "good" terms representative of information content. Various theories have been proposed for the identification of important natural language terms. The first, and best known of these is due to Luhn, and assumes that the value, or weight of a term assigned to a document or query

[+] Department of Computer Science, Cornell University, Ithaca, New York 14850

is proportional to the term frequency (TF), that is, to the number of times a term occurs in the text of a document, or document excerpt. [1] The Luhn theory reflects the fact that the use of high frequency terms is often essential for the specification of document content and for the retrieval of relevant information.

Unfortunately, the term frequency weighting does not always perform as expected. In particular, when few high frequency index terms are present in a given collection, or when the high frequency terms are evenly distributed across the documents — for example, when a given term occurs k times in each document — the upweighting of the high frequency terms will be of no avail. An alternative theory, proposed by Sparck Jones can then be used, which is based on the document frequency (DF) of a term, that is, on the number of documents in a collection in which a given term occurs. Specifically, it is suggested that terms with low document frequency are more important for retrieval purposes than those with high document frequency, because their comparative rarity will enhance their importance in query-document matching. If F_i is the document frequency of term i, then the inverse document frequency (IDF) weight J_i of term i is defined as

$$J_i = f(N) - f(F_i) + 1,$$

where N is the total number of documents in the collection and $f(x) = \lceil \log_2(x) \rceil$. [2]

When term frequency or inverse document frequency weights are used in a retrieval environment to distinguish good index terms from poor ones, it is found that they do not operate uniformly well across a number of different document collections. [3] The problem seems to be two-fold:

1) When term frequency weighting is used, it is likely that high weights will be assigned to high-frequency terms that have even frequency distributions across the documents of a collection — that is, they occur with approximately equal frequency in most of the documents; such terms are not useful in retrieval, since they cannot be used to distinguish the documents from each other.

2) When inverse document frequency weights are used, the highest weights will be assigned to those terms with the lowest document frequency, that is, DF = 1. Many of these terms will have a total frequency equal to 1, that is they occur only once in a single document. Such terms will account for very few matches between query and document terms, and will therefore not be very useful in a retrieval environment.

The foregoing seems to suggest that global frequency characteristics, incorporated in the TF or IDF weighting systems are too coarse, and should be replaced by frequency distribution characteristics which take into account the distribution of the term frequencies across the documents of a collection. Specifically, the following conjectures may be made concerning the importance and value for retrieval purposes of various types of index terms:

1) Terms with a skewed frequency distribution (which occur frequently in some documents, and rarely, or not at all, in some others) are preferred over terms with even (flat) distributions, since the latter are not useful in discriminating among the documents.

2) Terms with medium total frequency are preferred over terms with either very high, or very low total frequency, since very high frequency terms are likely to occur in all documents (even when they have fairly skewed frequency distributions), while very low frequency terms account for too few query-document term matches to be of importance.

The term discrimination model previously introduced ranks the documents in accordance with these criteria [4,5]:

a) medium frequency terms with skewed distributions;

b) low frequency terms with skewed distributions;

c) high frequency terms with skewed distributions;

d) terms with even (flat) distributions.

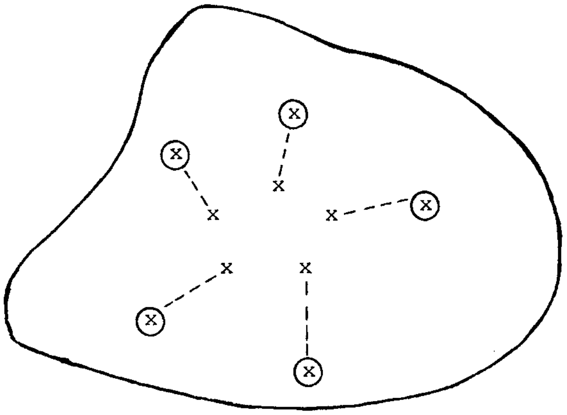
Specifically, the discrimination value DV_k of term k may be defined as $Q_k - Q$, where Q is the average pairwise document similarity for all document pairs in a collection, and Q_k is the same function computed after removing term k as an index term from all documents to which it is assigned. Obviously, if term k is a good discriminator, that is, if its presence helps in distinguishing one document from another, removing it as an index term will render the documents more similar to each other (because assigning it, will render the documents less similar) thus $Q_k > Q$. The reverse obtains when term k is a poor discriminator.

Fig. 1 is an illustration using five documents each denoted by an x. The distance between two x's is inversely related to the similarity between them, that is, the closer two x's in the Figure, the more similar are their content indicators. Assigning a good discriminator (or removing a poor discriminator from the content description) will render the documents less similar to each other; hence the inter-document similarity will decrease as shown in the Figure. Fig. 2 summarizes the computation of the term discrimination values for a given document collection.

The ten best discriminators obtained for a collection of 425 articles in world affairs from the 1963 issues of Time magazine are shown in Table 1. The ten worst discriminators are similarly shown in Table 2. For each term, the document and total frequencies are shown together with the corresponding frequency distribution, the discrimination value (that is, the amount by which the interdocument similarity is increased, or decreased, when removing the given term), and the rank of the term in discrimination value order. The following data are immediately obtainable from Tables 1 and 2:

a) the good discriminators have a average document frequency of about 20 (about 5% of the total number of documents), an average total frequency of 130 (30% of the number of documents) and there are approximately as many term occurrences of very low frequency 1-3 as there are of frequency greater than 3;

b) the poor discriminators, ranked 7560 to 7569 out of a total of 7569 terms, have an average document frequency of about 150 (about 45% of total number of documents), an average total frequency of 480 (about 115% the number of documents), and there are 5 times as many low frequency term occurrences of 1-3 as there are of frequency greater than 3.



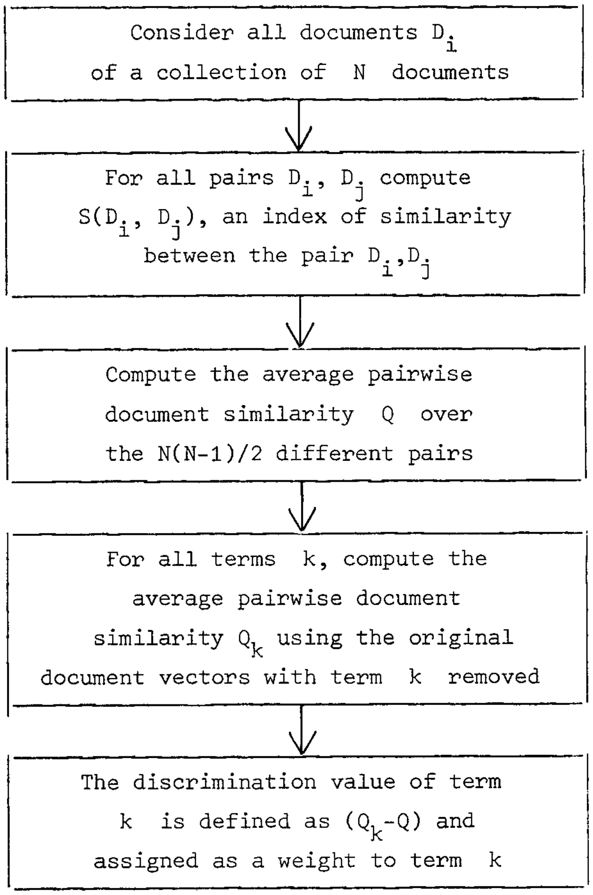
x Original Documents

⊗ Documents Following Assignment of Discriminator (or Removal of Nondiscriminator)

Illustration of Term Discrimination Model

Fig. 1

Obviously the poor discriminators have higher document and total frequencies than the good ones — each of them occur in almost half the documents of the collection — and the frequency distribution is in each case much flatter, most of the term occurrences being of frequency 1, 2, or 3. This suggests that bad terms might be transformed into good ones by reducing the corresponding document frequencies and producing skewed frequency distributions. A procedure for this purpose is outlined in the next section.



Consider all documents D_i of a collection of N documents

For all pairs D_i, D_j compute $S(D_i, D_j)$, an index of similarity between the pair D_i,D_j

Compute the average pairwise document similarity Q over the N(N-1)/2 different pairs

For all terms k, compute the average pairwise document similarity Q_k using the original document vectors with term k removed

The discrimination value of term k is defined as (Q_k-Q) and assigned as a weight to term k

Computation of Term Discrimination Values

Fig. 2

III. The Construction of Effective Vocabularies

It is seen in Table 2 that the nondiscriminators occur in many of the documents of a collection. It is then very likely that each document will contain several such nondiscriminators, and that many documents can be found that are jointly assigned a given group of nondiscriminators. The frequency of occurrence of the nondiscriminators may then be reduced by replacing each group of nondiscriminators by a single term which will then necessarily exhibit a much lower document frequency.

Specifically, the following process is suggested [6]:

a) the set of nondiscriminators for a given collection (with negative discrimination values) is clustered using one of the standard term clustering methods [5];

b) for each cluster of n nondiscriminators T, n+1 new terms T^1 are defined; term T^1_{n+1} is assigned to all documents originally containing all n terms T in the cluster; for each T_i, a new T_i^1 is assigned to all documents originally containing T_i but not containing the complete cluster.

The procedure is described in Fig. 3, and an

Term	Doc. Freq.	Total Freq.	Number of Documents in which Term Occurs with Frequency i 1-3	4-6	7-9	10-30	30+	Disc. Value	Rank
BUDDHIST	20	160	9	2	4	3	2	.24746	1
DIEM	23	183	9	4	4	5	1	.15565	2
LAO	14	92	7	1	2	4	0	.13799	3
ARAB	39	242	25	3	5	5	1	.12827	4
VIET	41	267	14	13	6	8	0	.11975	5
HURD	5	37	4	0	0	0	1	.11902	6
WILSON	16	78	9	4	0	3	0	.11597	7
BAATH	14	117	4	4	3	2	1	.10874	8
PARK	27	68	21	4	1	2	0	.10284	9
NENNI	10	61	5	1	1	3	0	.10156	10

Frequency Characteristics of Good Discriminators

(425 articles in world affairs from Time)

Table 1

Term	Doc. Freq.	Total Freq.	Number of Documents in which Term Occurs with Frequency i 1-3	4-6	7-9	10-30	30+	Disc. Value	Rank
WORK	151	300	138	9	1	3	0	- .33463	7560
LEAD	172	339	152	16	2	2	0	- .42653	7561
RED	139	382	108	21	6	4	0	- .47833	7562
MINISTER	170	385	141	20	5	4	0	- .56391	7563
NATION'S	201	441	172	24	1	4	0	- .73965	7564
PARTY	170	471	129	27	9	5	0	- .94874	7565
COMMUNE	189	508	148	30	6	4	0	- .98446	7566
US	174	656	120	26	13	-14	1	-1.13368	7567
GOVERN	242	677	192	29	10	11	0	-1.82873	7568
NEW	271	626	228	31	8	4	0	-1.86213	7569

Frequency Characteristics of Poor Discriminators

(425 articles in world affairs from Time)

Table 2

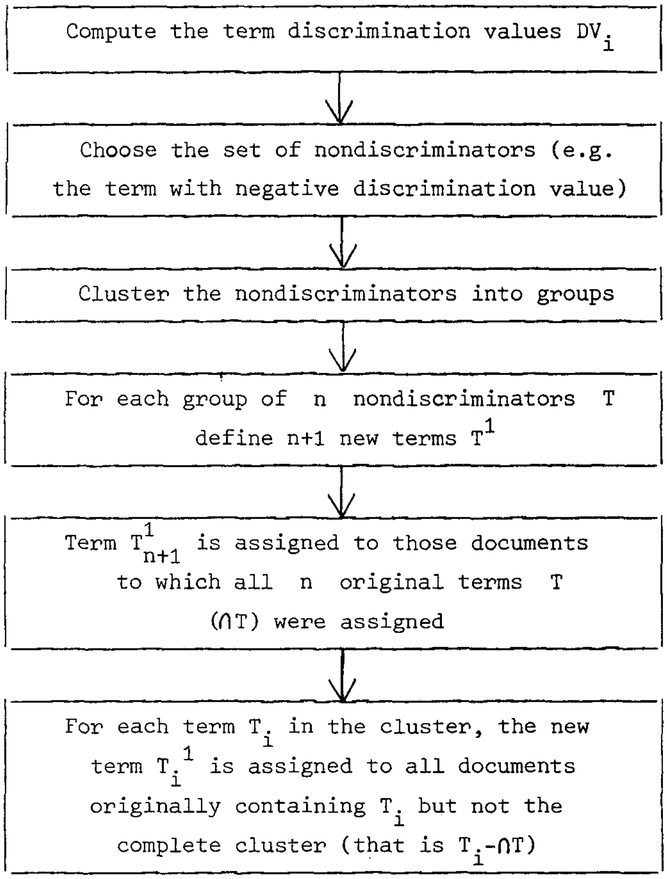
illustration is given in Fig. 4 for a term cluster of three terms T_1, T_2, and T_3.

It is seen in Fig. 4 that the complete cluster $\cap T$ was originally assigned to documents ①, ②, ③ and ④. Each of these items is now assigned a single new term T_4^1, replacing the three original terms. The other terms T_i^1 are defined from T_i by assigning them to the documents originally containing $T_i - \cap T$. In the illustration, the average document frequency of seven is reduced to three. (An alternative strategy in which a new term is defined from all possible intersections of old terms, that is, T_1 alone, T_2 alone, T_3 alone, $T_1 \cap T_2$, $T_1 \cap T_3$, $T_2 \cap T_3$, etc., is not useful because too many new terms are then defined each of which exhibits a very low occurrence frequency).

The foregoing strategy for the modification and redefinition of nondiscriminators was used experimentally with document collections in medicine (MEDLARS) and aerodynamics (CRANFIELD). In the former case, 160 nondiscriminators were clustered into seven groups of about 20 terms each, whereas for the aerodynamics collection, the 73 nondiscriminators were separated into five classes of about 15 terms each. The recall-precision output is presented in Fig. 5, averaged over 29 and 155 user queries, respectively.*

A standard word stem vocabulary in which word stems extracted from document abstracts are used as index terms is compared with the discriminator model. Two term discrimination strategies are used: in the first case, the nondiscriminators

*Recall and precision are well-known parameters to evaluate retrieval effectiveness. The curves closest to the upper righthand corner, where recall and precision are maximized represent the best performance.


Compute the term discrimination values DV_i
Choose the set of nondiscriminators (e.g. the term with negative discrimination value)
Cluster the nondiscriminators into groups
For each group of n nondiscriminators T define n+1 new terms T^1
Term T^1_{n+1} is assigned to those documents to which all n original terms T ($\cap$T) were assigned
For each term T_i in the cluster, the new term T_i^1 is assigned to all documents originally containing T_i but not the complete cluster (that is T_i-$\cap$T)

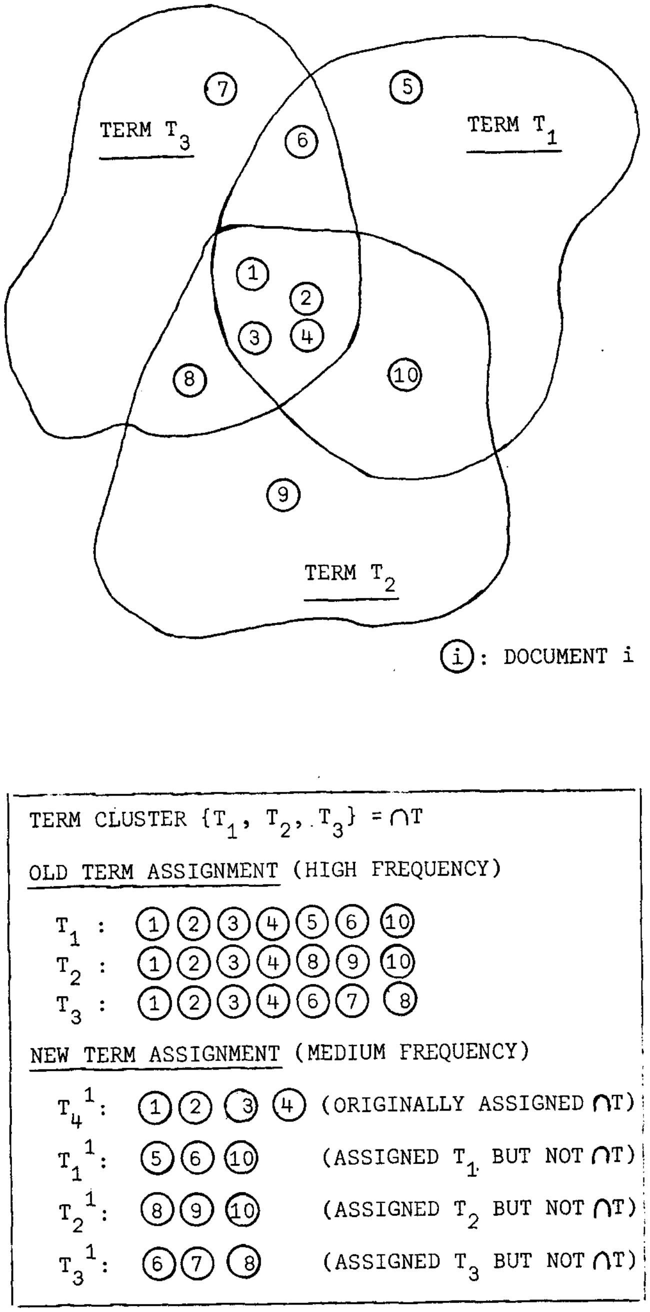
Transformation of Nondiscriminators

Fig. 3

are simply deleted from the vocabulary; this produces a significant improvement for the medical collection (Fig. 5(a)), but does not affect the aerodynamics vocabulary where the number of nondiscriminators is fairly small. In the second strategy, the nondiscriminators are clustered, and new terms with better frequency characteristics are created, as explained previously. It is seen that the latter strategy produces an improvement exceeding twenty percent in precision at most recall points for the two collections.

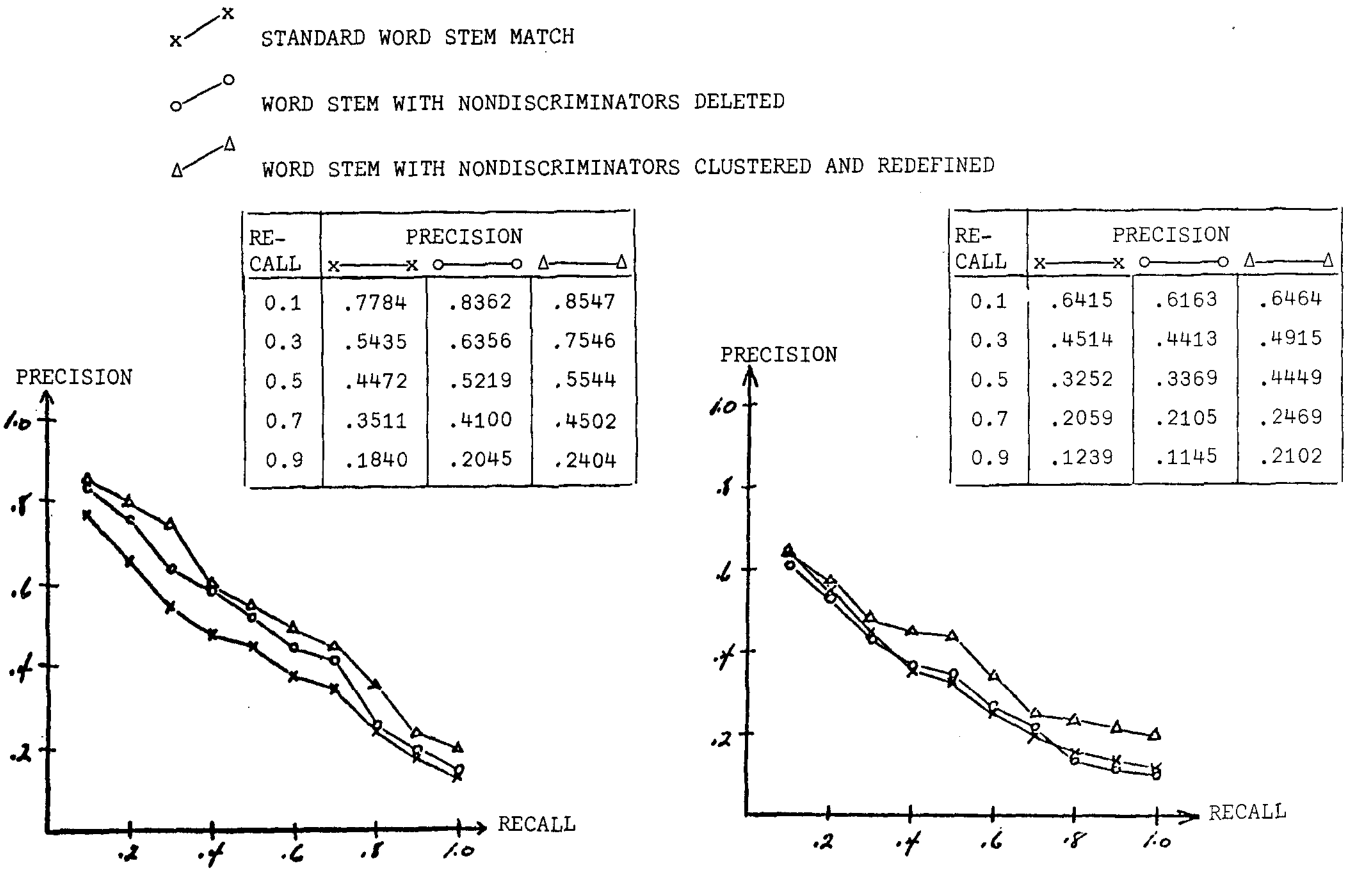
It is too early to specify in detail the frequency characteristics of an optimum vocabulary for each given subject area and user environment. Questions also arise about the stability of the term characteristics in a dynamic situation when many new documents are added, and old ones are deleted.

In a relatively static collection environment, however, it appears that effective automatic term grouping methods can be found to improve the natural language vocabularies used for indexing purposes. These methods are not only cheaper than the classical language analysis procedures based on syntax and semantics, but in a document retrieval application, they are also much more effective.


TERM T_3
TERM T_1
TERM T_2
7
5
6
1
2
3
4
8
10
9
i : DOCUMENT i
TERM CLUSTER $\{T_1, T_2, T_3\}$ = $\cap$T
OLD TERM ASSIGNMENT (HIGH FREQUENCY)
T_1 : 1 2 3 4 5 6 10
T_2 : 1 2 3 4 8 9 10
T_3 : 1 2 3 4 6 7 8
NEW TERM ASSIGNMENT (MEDIUM FREQUENCY)
T_4^1: 1 2 3 4 (ORIGINALLY ASSIGNED $\cap$T)
T_1^1: 5 6 10 (ASSIGNED T_1 BUT NOT $\cap$T)
T_2^1: 8 9 10 (ASSIGNED T_2 BUT NOT $\cap$T)
T_3^1: 6 7 8 (ASSIGNED T_3 BUT NOT $\cap$T)

Reassignment of Terms

Fig. 4



RE-	PRECISION		
CALL	x——x	o——o	Δ——Δ
0.1	.7784	.8362	.8547
0.3	.5435	.6356	.7546
0.5	.4472	.5219	.5544
0.7	.3511	.4100	.4502
0.9	.1840	.2045	.2404

RE-	PRECISION		
CALL	x——x	o——o	Δ——Δ
0.1	.6415	.6163	.6464
0.3	.4514	.4413	.4915
0.5	.3252	.3369	.4449
0.7	.2059	.2105	.2469
0.9	.1239	.1145	.2102

a) MEDLARS COLLECTION
(450 documents, 29 queries)

b) CRANFIELD COLLECTION
(424 documents, 155 queries)

Recali-Precision Output for Modified Indexing Vocabulary

(adapted from [6])

Fig. 5

References

[1] H.P. Luhn, A Statistical Approach to Mechanized Encoding and Searching of Literary Information, IBM Journal of Research and Development, Vol. 1, No. 4, October 1957, p. 309-317.

[2] K. Sparck Jones, A Statistical Interpretation of Term Specificity and its Application to Retrieval, Journal of Documentation, Vol. 28, No. 1, March 1972, p. 11-20.

[3] G. Salton and C.S. Yang, On the Specification of Term Values in Automatic Indexing, Computer Science Technical Report No. 73-173, Cornell University, June 1972, to appear in Journal of Documentation.

[4] K. Bonwit and J. Aste Tonsman, Negative Dictionaries, Scientific Report No. ISR-18, Section VI, Dept. of Computer Science, Cornell University, October 1970.

[5] G. Salton, Experiments in Automatic Thesaurus Construction for Information Retrieval, Information Processing-71, North Holland Publishing Co., Amsterdam, 1972, p. 115-123.

[6] S. Herzog and H. Kargman, Modification and Combination of Nondiscriminating Concepts in a Document Collection, Term Report CS 435, Dept. of Computer Science, Cornell University, Ithaca, May 1973.

QUESTIONS

Robert Brown:

In your automatic procedures, you might combine "red work". This seems on viewing your list of undesirable terms a possibility. Yet nobody is going to ask for documents about "red work".

Salton:

If these two terms happen to co-occur in a high number of documents and queries, then they might be placed together. If you say they are not good terms or a good phrase, then presumably the users will not use them. I am not interested in designing indexing vocabularies and then examining them to ask do they make sense. That procedure doesn't make sense to me at all. Human beings are very poor judges as to what is a good term and what is a bad term. I am interested in things that work in practice and result in good information retrieval. If the term "red work" is not a good term, then it will not be used in query formulation. So what is the difference? But I can show you improvements on the order of 20-30% from using the phrasing procedure.

William B. Malthouse:

Along the same line, when you are forming these phrases, do you form word order adjacency information; "Programming Languages" is an example of a phrase, but is your system just as likely to have made "Languages Programming" as a phrase?

Salton:

We have not introduced any constraints with respect to positioning. We are using simply an automatic clustering procedure to form the phrases, and clustering simply means that the terms co-occur with a given frequency. I simply know that the terms we break apart to form phrases are bad terms, and the phrase more likely than not is better than before. It is conceivable that you can improve the performance by restricting the phrase to the terms that occur in the right word order or to those which are syntactically related. But to write a syntactic analyzer costs you a great deal.

Tom Kibler:

It seems like the notion of meaningfulness should enter. For example if you have an article on archery in which "apple" occurs 25 times and "archery" occurs none at all, then "apple" is a good numerical separator for this article. However, it will never come up that "apple" is quite the right search word for that article. How do we get this jump into picking up articles that really do not contain the proper search word?

Salton:

This is one of those questions one gets all the time. And it is a question that is offered from the perspective of perfection. You argue that if that situation occurs then you do not get what you expect. And I agree entirely that you probably would not. But you see that they are many documents that are not retrievable. Typically we operate at a recall value of sixty percent meaning that we retrieve approximately 60% of that which we want to retrieve. Obviously we are never going to achieve perfection, but we do have evidence to suggest that with automatic procedures we do achieve a better performance than with systems using manual techniques. And on the average we do a lot better, for example 30% better, than MEDLARS.

Tom Kibler:

But it seems that you are retrieving a particular document. You are retrieving an empirical document, that is a document about a subject. With a theoretical document often the subject is not mentioned in the paper. And in fact, these documents often talk about a subject without even naming a subject specifically.

Salton:

It is conceivable to me that you might be right, but I see no evidence to support your view.

Richard Shrager:

In combining terms for example "programming" and "language", do you also include the information that these terms occur separately? If you do not, I am thinking you might be running the risk of throwing away documents that contain information on some subjects such as "programs that interpret languages", "programs that translate languages", etc.

Salton:

That is a good point. We can do either. We can add the phrases or replace phrases. We find that it works better if we do not remove any terms. The fact is that when you remove high frequency terms you often lose in recall, and when you remove low frequency terms, you often lose in precision. Our experience indicates that on the average you are better off to add phrases to the original.

Robert A. Gaskill:

It seems that you could make a good case for the high frequency terms by using them as negative discriminators. Have you looked into this possibilty?

Salton:

I take it that you are suggesting that we use the discrimination value as a weight. And the high frequency terms will have negative discrimination values, as they will in fact.

Robert A. Gaskill:

No, what I mean is that you want all documents that meet some criteria and do not have an occurrence of this term.

Salton:

No, that we do not do. We use a vector matching rather than a Boolean term matching. We do use the discrimination value as a weight; and those terms who have high frequency and negative discrimination value can cause an effect somewhat similar to what you describe.

DISCUSSION SESSION

Edward McCreight:

Is it known whether terms that belong together logically, in some sense, actually appear together contiguously? I ask this because a new algorithm has appeared that has a phenomenal performance on contiguous items.

Salton:

I do not have any data that gives an indication of how often terms that "belong together" appear contiguously. My own opinion is that you cannot make a very good case for that phenomenon. In our syntactic analysis work a few years ago we found that for every good case that we uncovered (where the terms were syntactically related), we found several bad cases. In general, the problem was that the syntax analyzer would not cause the assignment of the phrase to be made when actually it should. This does not mean that your conjecture is wrong; I simply do not know of any substantiating data. In general, the syntactic approach based on word ordering or proximity, seems to do well for increasing precision, but the recall performance suffers.

Edward McCreight:

The algorithm to which I refer is Wiener's substring matching algorithm, i.e., Peter Wiener, formerly of Yale University and now at the Rand Corp.

Salton:

One of the problems with Wiener's algorithm is that not only must you have the correct substring to cause a match but the precise ordering must also be given.

Leo Bellew:

In your presentation, you talk about terms and seem to ignore the concept of relations, but then you seem to bring it back with the idea of thesaurus and clustering. Then you say, "Well, that didn't work because the matching with the queries didn't work out well." Can you tell us how you have resolved this problem?

Salton:

What we do is to cluster the high frequency nondiscriminators automatically; we then restrict the terms that appear in the cluster to those that also appear in the set of queries. You see we are starting both with a document set and with a query set. Our latest algorithm works quite simple mindedly. We rank the nondiscriminators in ascending order in terms of discrimination value. We then take this set of nondiscriminators, appearing in the query set, and divide them into sets of triples, i.e., the first three, the second three, etc. We then assign the triples which we call T_{123} and T_{456} etc., to each document where they occur. We then divide each triple into its three pairs, i.e., T_{123} produces T_{12}, T_{23}, and T_{13}. We can then work an assignment process which allows us to assign singles, pairs, and triples; or we can assign pairs, pairs and triples, ignoring the single terms that cause the production of these. Our experience is by clustering nondiscriminators in this manner, the pairs and triple combination gives us a good performance. We also do a similar process with the high frequency terms (the good terms). But with the good discriminators we use an SPT system, i.e., we use singles, pairs and triples. Consequently, we use a combined system with the SPT for the good discriminators and the PT for the bad discriminators.

Patrick Mitchell:

When the system encounters a query with a large number of terms, is it going to have to take all combinations of those terms in all orders in order to test it against your term dictionary.

Salton:

No, not at all. We use a numeric encodement of terms so that the document is indexed by a series of numerical values (integers) so that the pairs and triples are combinations of numeric values. A pair becomes another numeric value, and likewise a triple is another separate numeric value also.

John T. Dockery:

To get an idea of how fast this runs, let me postulate a particular situation. If you were to select from an incoming message stream describing a constantly changing situation, could you construct a daily file structure based on a sampling of those messages?

Salton:

That is a very good question, and I wish I knew if we could. Currently, I just do not. To give you some idea of how our system works, we begin with a sample set of documents and a sample set of queries. We use those to construct our indexing vocabulary. We then use another part of essentially the same collection as the testing sample. We are operating in a University environment with small document collections, and the amount of computation to do what I have described is very little. I would suspect that we can do what you suggested, but I do not dare say that we can for a given cost.

Richard L. Guertin:

You stated in your presentation that you do not use logical operations. What about systems that do use logical operations. What about systems that do use logical operations and index only individual words or simple terms. In some cases these logical operations are implicit and either defined by the system itself or by the data base. Aren't those systems as good or better since they do not have to store duplicate information? It seems that your data base could grow quite large with redundant information.

Salton:

What I said is that we do not form Boolean queries. This is because once the Boolean expression is defined, then the system must assume that you want all documents that correspond to or satisfy that exact expression. We do a vector match of the composite terms for the set of given documents and then rank order the documents in terms of their match against the query. Our output consists then of a ranked list of documents in terms of their similarity to a query. This allows you to choose a threshold above which you will take all documents having this particular correlation value.

Richard L. Guertin:

Typical systems which index on every word of the title double or triple the size of the data base.

Salton:

Would you explain what you mean by double or triple the size of the data base.

Richard L. Guertin:

You add one record to the data base and the index increases proportionately either by a factor of one or two.

Salton:

When you say increase the size of the data base, then you actually mean the index?

Richard L. Guertin: Yes, So what is the rate of increase in the size of the index?

Salton:

I really cannot tell you. I could give you an idea, perhaps, by using the Time collection as an example. There are 7,500 single terms in the Time collection. The number of negative discriminators is of the order of 250. The number of terms with medium frequency having high discrimination value is of the order of 500. All the rest, namely 6,750 terms, are low-frequency terms that really make little difference whether you use them or not. We apply our procedures to the 250 negative discriminators and perhaps the top 250 of the high positive discriminators, and so the resulting increase in the number of terms is not all that great. One thing I should have mentioned in the presentation was that if you're interested in the boundaries between the good and poor discriminators, then this is mathematically defined in the work of Clement Yu and is contained in a Cornell technical report. This boundary separation, I will warn you, involves some rather heavy mathematics.

William Malthouse:

I noticed that your simple-minded clustering seems to work fairly well; have you also experimented with allowing the overlap, for example, T_{234} as defining the triple?

Salton:

At the moment this is what we are doing but we do not claim that it is optimal. It is quite conceivable that we could do better.

Bea Marron:

A simple question, the 7,500 index terms from the Time collection represented how many documents?

Salton: That was 425 full-text Time magazine articles.

Richard S. Marcus:

Intuitively, it seems that your low-frequency terms provide the greatest discrimination when they are used by the user. Is your reason for clustering them based on cost or efficiency rather than treating them as such?

Salton:

I agree with your premise. The discrimination model categorizes them as bad terms because their values are close to zero, which is to say if one looks at the document space, the inclusion of these terms has little effect. One neither causes a contraction nor a separation of the existing members of the document space. So that even if the user uses such a term, it will match so few documents in the collection as to cause very little effect. Nevertheless, I think to delete those terms would cause a significant loss in precision. So that rather than delete them we group them, because as you say when they are used they may be exerting an effect on precision.

Richard E. Nance:

I think that what may be bothering Richard Marcus is that recall/precision is a measure if you're interested in generic retrieval.

Salton: What do you mean by generic retrieval?

Richard E. Nance:

I mean when you're not trying to perform a specific search for a given document. But for the user who is interested in a specific search, maintaining these near zero discrimination terms might be beneficial. I agree with you, however, that for a generic search, it is probably not cost effective.

Salton:

If you're interested in high precision, then I would say you do nothing to those low-frequency terms and you restrict yourself to the right to left transformation. So, this is quite right. Remember, our measures of recall/precision are averaged over several queries, and they represent what the result will be for the "average user". Our combination of the two transformations is based on the premise that you want best average performance for the average user.

Blanton C. Duncan:

I apologize for not reading your paper; therefore, I have to ask this question. You have the full text of the section of Time magazine from 1963, and there is a section which mentions MEDLARS. Now what is the text on which that MEDLARS comparison is based?

Salton:

The text on which the MEDLARS comparison is based is largely an abstract of a MEDLARS document. You see, we do natural language analysis, so that we need more than titles. One of our people went down to the National Library of Medicine and we copied the first page of each document, and if there was an abstract we used it as our basis; if not, we used the first paragraph. We generally use abstract-length excerpts unless the full text is not exceedingly long.

Martin Dillon:

To what extent are SMART-type techniques making their way into operational document retrieval systems?

Salton:

Please do not call this SMART, because the people who funded us say that they have funded us long enough for SMART. If you ask, "What is the actual impact of these techniques in large operational retrieval systems?", then the answer is "They are fairly limited." Not even basic automatic indexing techniques are really being used. Still, there are bits and pieces that have been used in various places. For example, at Rome Air Development Center the Air Force is using a system which is a SMART-type system. At EURATOM, they are using our relevance feedback techniques. Numerous small companies have implemented systems that resemble SMART and use those techniques. However, if you look at the very large systems, such as RECON, then the impact has not been to a great extent. One of the serious drawbacks to the advance and the use of automatic techniques is the difficulty in data entry.

A. A. Brooks: Do you preserve word order in the formation of pairs?

Salton: No.

A. A. Brooks:

What type of query is this? Is this a one-shot query or feed-back improved query? And would you expect a significant difference in performance if feedback were allowed in the query formulation?

Salton:

Our work is done with the query that is a single statement, perhaps in English, and does not utilize feedback. However, we do use relevance feedback techniques in reformulating the query.

William Malthouse:

I notice that in your slides the performance improvement seems to be a linear shift of basically a linear relation. It seems to me that you would be wanting more of a shape change to produce a convex curve.

Salton:

I have no idea what kind of shift it is because that is not the way we view it. Our particular view is that, given an identical recall value for two techniques, then if one gives a precision of .3 and the other a precision of .5, we have had an improvement of .2 in the precision of one technique over another.

William Malthouse:

What I was saying is for the high recall user. You were saying that the percentage improvement in precision at high recall is the same.

Salton: I don't think that is the case.

John T. Dockery:

In your 1968 text, you described multiple strategies for retrieval, and did you not have recall/precision curves that actually change shape based on multiple passes?

Salton: Yes, by multiple pass you mean feedback in query formulation?

John T. Dockery:

Correct, and what is shown here seems to be based on a single query statement.

Salton: Yes, for the first couple of feedback steps we got considerable improvement.

Nagib Badre:

You mentioned that syntactic and semantic analysis may not buy you that much when you're interested in high recall. I may not be familiar with the literature, but are there any comparison studies showing that result both for high recall and for high precision situations? Have you made any such studies or do you know of any?

Salton:

Are you asking if anyone claims that syntactic analysis is good for certain purposes, such as high precision?

Nagib Badre:

You say that it does not buy you much, and I am wondering if there are any specific studies that support that?

Salton:

There have been specific studies by us and others that, taking an evaluation view point of improvements for the average user, show the syntactic approach to give a lower performance value. This does not mean that if you took the attitude of being interested in only high precision, that is, having a very low recall value but having all documents retrieved being highly relevant, that syntax would not prove useful and actually improve performance. And there is a whole set of people who claim that syntax is necessary because of the ambiguity of the natural language, but those statements are largely sentiment rather than fact.

Nagib Badre: There have been actual studies to this effect?

Salton:

I do not know of any system where a syntax analyzer has been used and improvement has been obtained; however, note that the syntax analyzers that can be incorporated into a computational system are simple in nature, generally Chomsky type 2, context-free analysis. It is conceivable to me that a day will come when syntax analyzers with sufficient accuracy and ease of use will be made available. Then it is possible that for high precision users improvement will be obtained. You still will not get improvement if you take the view of the average user.

Robert Brown:

What do you do when you make your right to left translation, for example, to make sure that the resulting phrases make any sense with respect to a natural language query?

Salton: I don't know what you mean by "do they make any sense"?