A Note on Term Weighting and Text Matching

Gerard Salton
Chris Buckley*

TR 90-1166
October 1990

Department of Computer Science
Cornell University
Ithaca, NY 14853-7501

*This study was supported in part by the National Science Foundation under Grant IRI-89-15847.

A Note on Term Weighting and Text Matching

Gerard Salton and Chris Buckley *

October 29, 1990

Abstract

In information retrieval, it is not uncommon to be faced with large collections of unrestricted natural-language text. In such circumstances, the text analysis and retrieval operations must be based mainly on a study of the text collections actually under construction. Two main operations are of interest: a text analysis operation designed to assign content identifiers to the stored texts, and a text comparison system designed to identify texts covering particular subject areas.

In the present note, some details are given concerning the usefulness of term weighting systems for the content analysis of natural-language texts, and of text matching strategies designed to identify relevant text items in answer to available search requests. A sample collection of electronic mail messages is used for experimental purposes.

1 Introduction

In a previous report in this series, text analysis and retrieval experiments were described designed to process large text collections in unrestricted discourse areas.[1] In such circumstances, one cannot rely on standard approaches to text analysis requiring complete specifications of the semantic characteristics of the discourse areas of interest. Instead, it is necessary to rely on a study of the texts actually used in the retrieval operations.

The retrieval methods introduced in the previous report were based on assignments of complex term weights to the text units of interest – for example, text paragraphs and text sentences – and on the use of global text matching operations between text excerpts of varying scope. When sufficient similarities

*Department of Computer Science, Cornell University, Ithaca, NY 14853-7501. This study was supported in part by the National Science Foundation under grant IRI 89 15847

were detected between particular text pairs, based on both global as well as local text matching characteristics, links were introduced between the corresponding text excerpts to indicate that the texts were appropriately related. Such text links could serve for selective text traversal by making it possible to follow the links from particular text excerpts to additional related ones. The placement of links between related text excerpts could also lead to improved retrieval operations by directly identifying sets of linked documents that would be jointly retrieved in answer to particular search requests.

The earlier report omitted details concerning the choice of term weighting functions and text comparison methods. These questions are briefly examined in the present note.

2 Term Weighting Systems

A standard method of text indexing consists in recognizing individual text words, eliminating common words included on a word-exclusion list, and using the remaining words, or word stems, for the content identification of the corresponding texts.[2] Since the text words are not all equally important for purposes of content representation, it is useful to assign importance factors, or weights, to the terms in decreasing order of their presumed importance for text content identification.

Assuming that t terms in all are used to index a document collection, each document D_i is representable as a vector of terms

$$D_i = (w_{i1}, w_{i2}, ..., w_{it}) \quad (1)$$

where w_{ik} represents the weight of term T_k in document D_i.

Several criteria must be taken into account in generating a term weighting function that would be useful for document indexing:[3]

a) The term frequency, tf, of a term, representing the occurrence frequency of a term in a given text, or text excerpt, is normally reflective of term importance. In general, the higher the frequency of occurrence of a term in a text, the more important the term becomes as an indicator of text content. Using the term frequency, tf_{ik}, for term T_k in document D_i as the basic measure, the tf factor can be computed in a number of different ways as follows:

$$b = 1.0; \quad t = tf; \quad n = 0.5 + 0.5 \, \frac{tf}{max\ tf}.$$

The last formula represents an enhanced term frequency measure in which the values are normalized to lie between 0.5 and 1.0.

b) In addition to the term frequency, an inverse collection frequency factor, known as idf, may be used to enhance the importance of terms assigned to few documents in a collection, while downgrading the terms occurring in many documents. Assuming that a collection of N documents is available of which n_k documents contain a given term T_k, the *idf* factor can be computed variously as follows

$$x = 1.0; \quad f = log N/n; \quad p = log\,\frac{N-n}{n}.$$

When both the term frequency as well as the inverse collection frequency are used to determine the weight of a term assigned to a document – for example, by multiplying the term frequency by the inverse collection frequency – the terms receiving the greatest weight are those occurring frequently in individual documents but rarely on the outside. Combined term frequency and inverse document frequency factors are known to provide a high-order of performance in actual retrieval environments.[3]

When the documents, or document excerpts, used as retrieval units are all of comparable length, and homogeneous in appearance (that is, when the documents consist of similar types of text) the computed "term frequency times inverse document frequency" ($tf \times idf$) weight can be used directly to form the document vectors of expression (1). In that case, the magnitudes of the occurrence frequencies, and of the collection frequencies of the terms assigned to different documents are directly comparable. In many text environments, it is however necessary to process documents of vastly different length. In that case, terms occurring in the longer documents may exhibit much higher frequency factors than terms assigned to shorter text items. A normalization factor may then be added to the tf and idf factors to reduce all documents to a common length. If w_k represents the combined ($tf \times idf$) weight of a particular term T_k in a document vector, the normalization could be computed as

$$x = 1.0, \quad or \quad C = \frac{1}{\sqrt{\sum_{vector}(w_i)^2}}$$

Using the three term-weighting components described earlier, the term weight assignment used in a particular collection environment can be specified by using two identifying triplets, the first one characterizing the weight used for the document terms, and the second the weight assignment used for query terms. The three components of each triplet specify term frequency, inverse collection frequency, and weight normalization factors, respectively. A weight specification such as ($tfc \cdot nfx$) thus indicates that normalized ($tf \times idf$) weights are used for document terms, whereas enhanced $tf \cdot idf$ weights without normalization are used for the queries.

The effectiveness of term weight assignments may be assessed by using a set of 1984 electronic-mail (e-mail) messages for experimental purposes. The e-mail collection represents heterogeneous materials of vastly different scope, ranging in length from one or two lines for certain messages, to many pages for others.

Many different writing styles are used to express message content, and the topic areas differ from message to message. 180 special messages are identified as information requests, and each such request, or query, is then compared with the complete message collection. Assuming that query as well as document messages are represented by sets of weighted terms, as in expression (1), a pairwise query-document similarity computation can be performed that generates a similarity coefficient between the respective texts as follows:

$$Sim(D_i, Q_j) = \sum_{k=1}^{t} w_{ik} \cdot w_{jk} \tag{2}$$

or

$$Sim(D_i, Q_j) = \sum_{k=1}^{t} min(w_{ik} \cdot w_{jk}). \tag{3}$$

The w_{ik} factors in expression (2) represent the term weights in the documents, and the w_{jk} factors represents the term weights in the query vectors. When a similarity coefficient is obtained between each stored document and each query, the stored items can be arranged in decreasing order of the similarity will the respective queries, and the top items retrieved early in a search can be submitted to the users as responses to the queries.

The effectiveness of the retrieval operations can be assessed by computing average recall and precision figures reflecting, respectively, the proportion of relevant documents retrieved in answer to each query, and the proportion of nonrelevant items that could be rejected. The recall and precision computations must be based relevance information for each document with respect to each query. For the collection of electronic mail messages, objective relevance indications can be extracted from the header information available with each message. Message headers normally provide information about senders and intended receivers of the messages, subject specifications, and possible references to other messages previously sent through the information network. For the experiments conducted with the sample message collection, a message was assumed to be relevant to a particularly query message if either the two messages headers carried an identical subject specification, or if a reference could be detected between a document header and the corresponding query header, indicating that the message may have been sent as a response to the query. Even though the relevance data provided by the message headers are far from perfect, the use of objective relevance information is much preferable to an elaborate subjective assessment of relevance of each document with respect to each query.

Table 1 contains a qualitative evaluation of several term weighting systems used for document term indexing. The number of asterisks reflects the overall performance of the corresponding term weight assignment in a retrieval setting. As the Table shows, the basic term frequency factor used without inverse collection frequency or length normalization is not very useful by itself. The performance improves when length normalization is used, as shown in the middle section of Table 1. The best performance is obtained with a normalized ($tf \times idf$) term weight assignment.

A more detailed evaluation for the better term weighting systems is shown in Table 2. The retrieval precision values given in the Table represent average precision values for three specific recall points (recall = 0.25, 0.50, and 0.75), averaged over the 180 queries in use with the message collection. Precision data are included in Table 2 for two types of message texts: the "quoted" and "unquoted" collections, respectively. In the former case, the full message text is used, including portions of text that may have been quoted from other messages. In the latter case, quotations from other message texts are removed prior to the text matching operations. When the quoted texts are considered, the detection of relevant documents containing quotations from the corresponding queries may be relatively simple. Not infrequency such documents are retrieved at the head of the list of retrieved items with perfect, or near-perfect query-document similarities. Such a retrieval performance is, however, not applicable to random text collections that lack the quoted text portions.

The performance figures of Table 2 were obtained by comparing each query with every document in the collection, arranging the documents in decreasing order of the query-document similarity, and computing the retrieval precision at recall levels ranging from zero (before any relevant item is retrieved) to 1 (after the last relevant item is retrieved). The current experiments utilize the inner minimum similarity function of equation (3) for the query-document comparison. Several dozen query and document term weight combinations were tried experimentally. Table 2 contains evaluation results for the more useful combinations. The following conclusions are evident:

a) Term frequency, or enhanced term frequency weights (t or n factor) should be used for both query and document terms.

b) The inverse collection frequency factor (f factor) should be used for the queries but not necessarily for the documents. The *idf* factor can be computed on the fly as each query is processed.

c) A length normalization factor (c factor) is essential for the document texts because of the large variations in message length. The normalization factor is not needed for the query texts.

The results of Table 2 indicate that nearly perfect results are obtainable for the quoted collection. In that case, the average precision reaches 88 percent for the 180 test queries, indicating that almost all relevant items are retrieved before the nonrelevant ones. The average precision values are not as impressive for the unquoted collection. The results do, however, indicate that large numbers of relevant documents are retrievable relatively easily even for the unquoted case. Because the inverse collection frequencies of the terms (the *idf* factors) vary continuously in situations where documents are constantly added to the collection, it is best to use the *idf* weights (the f factors) only for the queries, but not for the documents in dynamic collection environments. This leads to the following specification for an effective term weight assignment:

a) For document retrieval in dynamic collection environments where queries are compared with continuously changing document collections, use $(nxc \cdot nfx)$.

b) For text linking operations, where each document text is compared with all other texts in order to generate links between semantically similar texts, a single term weighting system must be used for all texts. Typically a text linking system is used for book-type materials where the collection composition does not change over time. In that case, use $(nfc \cdot nfc)$.

3 Composite Text Similarity Measures

It was noted earlier that global text comparisons may lead to the identification and retrieval of texts with superficially similar word patterns that are, however, semantically distinct.[1] For example, texts dealing with "army bases" might in some circumstances be confused with texts specifying "lamp bases" or "baseball bases". Such confusions may be avoided by carrying out text comparisons at various levels of granularity: for example, texts with global similarities might also be compared at the sentence-level, or the phrase level, to determine local similarities within particular sentences, in addition to the global similarities between the complete texts. When different texts exhibit both global, as well as local, similarities, the expectation is that the texts actually cover semantically related subject areas.

The use of local as well as global text matching systems leads to the use of composite text similarity measures. The global similarity of expression (2) is then enhanced for text pairs that also exhibit an appropriate number of local similarities. Two kinds on enhanced text similarity measured suggest themselves:

a) A similarity measure may be used that does not depend directly on the actual number of local sentence (or phrase) similarities. In that case an enhanced similarity coefficient "new sim" may be computed from the original similarity "old sim" whenever at least i matching sentence pairs are detected in a document pair with a minimum pairwise sentence similarity of j. The enhanced similarity is specified by the parameters NEED $(i, j, mult)$ and is computed as

$$new\ sim \ = \ old\ sim \ \cdot \ \frac{1}{mult} \tag{4}$$

where mult is an appropriate multiplier between 0 and 1.

b) Alternatively, an enhanced similarity measure could be used which does vary with the actual number of sentence-pair matches. This is known as ADD $(i, j, mult)$. The corresponding similarity is computed as

$$new\ sim \;=\; old\ sim\ (1 + [k - i] \cdot mult) \qquad (5)$$

where k is the actual number of matching sentence pairs exhibiting a minimum sentence similarity of j, and i is a parameter that adjusts the number of matching sentence pairs actually taken into account.

Table 3 shows average recall-precision results for the enhanced text similarities including both global text similarities as well as local sentence similarities, for the e-mail messages used with 180 queries. Results are shown for both the quoted and the unquoted collections. In addition to the three-point precision figures used earlier, recall and precision data are also shown in Table 3 for fixed retrieval thresholds of 15 retrieved items and 30 retrieved items obtained for each query. The precision output computed for the threshold 15 and threshold 30 output is not comparable with the three-point average precision figures, because the maximum attainable precision is necessarily limited when as many as 15 documents must be examined for each query regardless of the retrieval ranks of the relevant items. (Typically, only one or two relevant documents are present in the collection for each query, and these items are often retrieved early in a search; when a fixed retrieval threshold is used, large numbers of non relevant documents must then necessarily be included in the evaluation in addition to the few relevant items that may exist.)

The results of Table 3 show that the sentence match requirement makes very little difference for the quoted collection. When near-perfect retrieval performance is obtainable with the global text comparisons alone, no obvious need exists for calling on added search refinements. Some improvements are, however, evident when the sentence matching is used with the unquoted collection. The best performance for NEED (2, 2, 0.75) requires at least two matching sentence pairs with a sentence similarity of at least 2, in addition to the usual global text similarity. It was noted in the earlier report that the sentence similarity requirement operates as a precision device, in the sense that the number of texts satisfying the matching criteria is smaller when sentence similarities are used in addition to global text similarities. This depresses the recall and greatly enhances the precision. This effect is not easily noticed when average precision data are presented for three different recall values as the output of Table 3.

4 Failure Evaluation

The electronic mail environment used in this study is characterized by a number of special problems that are not always present in conventional text collections. These problems are due in part to the use of objective relevance judgments where items are treated as mutually relevant based on information included in the message headers. Various circumstances may then lead to depressed recall and precision values, even when the retrieval system operates flawlessly:

a) Documents retrieved early in a search that are actually relevant to a particular search request may be falsely labelled as nonrelevant because of slight variations in the available subject descriptions attached to the respective documents. Examples are

i)	query subject:	Modernizing Ada
	document subject:	Modernizing Ada
ii)	query subject:	Does Ada need multiple inheritance
	document subject:	Adding multiple inheritance
iii)	query subject:	$PA2$ disable
	document subject:	$PA2$ key

In each case, substantial query-document similarities are correctly obtained by the text retrieval system, but the false formal relevance information due to the differences in the subject specifications leads to depressed retrieval precision measures.

b) The reverse problem arises when chains of messages are present that all carry the same subject identification. Such messages are all treated as objectively relevant to a particular query. In actual fact, the initial document replying to the query may indeed be relevant to that query. Subsequent messages in the message chain may, however, refer not to the query at all, but to replies to the query, or to replies to replies to the query, all the while maintaining the same subject identification. Typically, the subject matter shifts from message to message and the later documents are not in fact relevant to the original query, although the system says they are. When these later documents are not retrieved, or are retrieved late in a search (as they should be), the recall performance is falsely depressed.
Since the objective relevance assessments are not under the control of the system designer, very little can be done in practice, short of using new, or different relevance data for the experiments.

The other main evaluation problem is due to the extreme variability between query and document lengths. Some messages are very short – one or two lines at most; other may be tutorials extending over several dozen pages. Text length normalization techniques can be used to reduce all texts to a common length as previously noted. However, difficulties arise nevertheless when certain texts are identified by only a few terms (the short ones), and other by many terms (the long ones):

c) When short documents or short queries are processed, it is often difficult to find enough matching terms to produce the required number of sentence matches. In such circumstances, a relevant document may not be retrievable in response to a query, even when a large proportion (but not a large number) of matching terms exist. This leads to depressed recall measurements.

d) The reverse problem arises when the query and document texts are very long. Here large global as well as local similarities may be detected, leading to an early retrieval of the corresponding documents. When such documents are not relevant, the precision performance is depressed.

Additional problems that may be hard to control in an e-mail environment arise when computer programs are included as part of the message texts. Certain program components may then lead to false text matches. This is true also when the message signatures are not correctly removed, and the signature components for different messages interfere with the text matching system.

In view of the difficulties presented by the objective relevance assessments and the extreme variations in message lengths, the performance data presented in Tables 2 and 3 may reflect a high standard of performance.

References

1. Gerard Salton and Chris Buckley, Flexible Text Matching for Information Retrieval, Technical Report 90-1158, Computer Science Department, Cornell University, Ithaca, NY, September 1990.

2. G. Salton and M.J. McGill, Introduction to Modern Information Retrieval, McGraw Hill Book Co., New York, NY 1983.

3. G. Salton and C. Buckley, Term-Weighting Approaches in Automatic Text Retrieval, Information Processing and Management, 24:5, 1988, 513-523.

	Document Weight	Weight Components	Rating
1.	Basic term frequency run (t x x)	tf	*
2.	Enhanced term frequency run (n x x)	enhanced tf	**
3.	Term frequency with cosine normalization (t x c)	tf, normed	***
4.	Enhanced term-frequency with cosine normalization (n x c)	enhanced tf, normed	***
5.	Term frequency times inverse document frequency normalized (t f c)	tf, idf normed	***
6.	Enhanced term frequency times inverse document frequency normalized (n f c)	enhanced tf idf, normed	****

Table 1: Qualitative Retrieval Effectiveness of Term Weighting Systems (E-Mail News Collection)

Term-Weight Combinations			Three-Point Average Precision	
Document Weight		Query Weight	Unquoted Collection	Quoted Collection
t x c	·	n f x	0.5612	0.8780
n x c	·	n f x	0.5883	0.8775
n x c	·	t f x	0.5594	0.8773
n f c	·	n f x	0.5580	0.8809
n f c	·	t f x	0.5479	0.8795

Table 2: Average Precision Data for Some Effective Term-Weight Assignments (1984 e-mail messages, 180 queries)

	Three Point Precision	Threshold 15 Recall	Threshold 15 Precision	Threshold 30 Recall	Threshold 30 Precision
1. Quoted Collection					
Basic nfc-nfc (no sentence matches used)	0.8802	0.9156	0.1659	0.9291	0.0869
Need (1, 2, 0.75)	0.8779 (-0%)	0.9130	0.1656	0.9321	0.0878
2. Unquoted Collection					
Basic nfc-npc (no sentence)	0.5600	0.7545	0.1326	0.8033	0.0731
Need (2, 2, 0.75)	0.5801 (+ 3.6%)	0.7483	0.1348	0.8041	0.0744
Need (2, 3, 0.75)	0.5623 (+ 0.4%)	0.7367	0.1300	0.8021	0.0735
Add (0, 2, 0.10)	0.5633 (+ 0.6%)	0.7218	0.1315	0.8038	0.0743
Add (0, 3, 0.10)	0.5674 (+ 1.3%)	0.7324	0.1307	0.8067	0.0743

Table 3: Evaluation Results for Composite Text Similarity Measurements (Global plus Local Similarities) (1984 e-mail messages, 180 queries)