ORNL/TM-13756

Computer Science and Mathematics Division

NEW TERM WEIGHTING FORMULAS FOR THE VECTOR SPACE METHOD IN INFORMATION RETRIEVAL

Erica Chisholm[1] and Tamara G. Kolda[2]

Computer Science and Mathematics Division
Oak Ridge National Laboratory
Oak Ridge, TN 37831–6367

[1] Email: erica@msr.epm.ornl.gov
[2] Email: kolda@msr.epm.ornl.gov

Date Published: March 1999

Research supported by the Applied Mathematical Sciences Research Program, Office of Energy Research, U.S. Department of Energy, under contracts DE–AC05–96OR22464 with Lockheed Martin Energy Research Corporation and by the Office of Science, U.S. Department of Energy and administered by the Oak Ridge Institute for Science.

Prepared by
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee 37831-6285
managed by
LOCKHEED MARTIN ENERGY RESEARCH CORP.
for the
U.S. DEPARTMENT OF ENERGY
under contract DE–AC05–96OR22464

Contents

List of Tables

NEW TERM WEIGHTING FORMULAS FOR THE VECTOR SPACE METHOD IN INFORMATION RETRIEVAL

Erica Chisholm and Tamara G. Kolda

Abstract

The goal in information retrieval is to enable users to automatically and accurately find data relevant to their queries. One possible approach to this problem is to use the vector space model, which models documents and queries as vectors in the term space. The components of the vectors are determined by the term weighting scheme, a function of the frequencies of the terms in the document or query as well as throughout the collection. We discuss popular term weighting schemes and present several new schemes that offer improved performance.

1. Introduction

Automatic information retrieval is needed because of the volume of information available today — there is too much information to be indexed manually. Most people have used some type of information retrieval system in the form of Internet search engines. Search engines are based on information retrieval models such as the Boolean system, the probabilistic model, or the vector space model [7].

We focus on the vector space model, described in Sect. 2, which models documents and queries as vectors and computes similarity scores using an inner product. The performance of the vector space model depends on the *term weighting scheme*, that is, the functions that determine the components of the vectors [9]. In Sect. 3, we outline the ideas underlying term weighting and present several popular term weight schemes. In Sect. 4, we describe new term weighting formulas. Sect. 5 gives experimental results comparing the old with the new, and Sect. 6 concludes that the new methods are an improvement over existing schemes.

2. The Vector Space Model

In the vector space model, individual documents and queries are represented as vectors in term space. The term list for a given document collection is compiled as follows. Words appearing in only one document are removed. Numbers, punctuation, and stop words are also removed. The remaining words form our set of terms. (See Kolda [7] for further details on preprocessing.)

To compare a document and query, we find their *similarity score* by computing their dot product. For example, Table 1 shows two partial documents from MEDLINE, and Table 2 shows a query from the MEDLINE test collection.[1] (See the Appendix A for the complete documents and query from this table.) Note that the vectors are sparse because only a few of

[1] We cannot show all terms in each document for lack of space, but all terms in common to the example query are given.

Document 1			Document 2		
Term ID.	Word	Weight	Term ID.	Word	Weight
37	accompany	0.09	63	acids	0.07
572	blood	0.36	1341	determined	0.07
1034	content	0.18	1899	fatty	0.07
1925	fetal	0.09	1925	fetal	0.44
2051	free	0.09	1930	ffa	0.22
2559	infant	0.18	2051	free	0.07
2876	levels	0.09	2125	glucose	0.29
3718	placenta	0.09	2876	levels	0.29
⋮	⋮	⋮	⋮	⋮	⋮

Table 1: Documents from the MEDLINE test collection.

Query		
Term ID.	Word	Weight
59	acid	0.30
63	acids	0.30
494	barrier	0.30
1899	fatty	0.60
1926	fetus	0.30
2876	levels	0.30
3358	normal	0.30
3718	placenta	0.30

Table 2: Query from the MEDLINE test collection.

all possible words appear in any one document, so we show only the nonzero entries and their indices. Here, the common terms between Document 1 and the query are "levels" (2876) and "placenta" (3718), and the similarity score of Document 1 is

$$(0.09 \times 0.30) + (0.09 \times 0.30) = 0.05 \ .$$

The common terms between Document 2 and the query are "acids" (63), "fatty" (1899), and "levels" (2876), and the similarity score of Document 2 is

$$(0.07 \times 0.30) + (0.07 \times 0.60) + (0.29 \times 0.30) = 0.15 \ .$$

Because Document 2 has a higher similarity score than Document 1, Document 2 would be retrieved before Document 1. The *weights* in the preceeding examples were computed by using a normalized frequency vector. For example, the term "acids" appears once in Document 2 and the norm of the vector of the frequencies is $2\sqrt{47}$, so the weight for "acids" is $1/2\sqrt{47} = 0.07$. This is a relatively simple weighting scheme; we discuss several others in the next two sections.

It is useful to give a geometric interpretation to the vector space notion of "similarity." Consider the dot product equation

$$\cos\theta = \frac{d \cdot q}{||d||||q||} \ ,$$

where d is a document vector, q is a query vector, and θ is the angle between them. If d and q are normalized so that their magnitudes are one (as in the previous example), the preceeding equation then reduces to $\cos\theta = d \cdot q$, so the similarity score is a measure of cosine of the angle between the vectors. If we rank the documents according to their similarity score from highest to lowest, the highest scoring document has the smallest angle between itself and the query. In the previous example, the angle between the document and the query vectors is approximately $77°$.

3. Term Weighting

Proper term weighting can greatly improve the performance of the vector space method. A weighting scheme is composed of three different types of term weighting: local, global, and normalization. The term weight is given by

$$L_{ij} G_i N_j \ ,$$

where L_{ij} is the local weight for term i in document j, G_i is the global weight for term i, and N_j is the normalization factor for document j. Local weights are functions of how many times each term appears in a document, global weights are functions of how many times each term appears in the entire collection, and the normalization factor compensates for discrepancies in the lengths of the documents.

Formula	Name	Abbr.
$\begin{array}{ll} 1 & \text{if } f_{ij} > 0 \\ 0 & \text{if } f_{ij} = 0 \end{array}$	Binary	BNRY
f_{ij}	Within-document frequency	FREQ
$\begin{array}{ll} 1 + \log f_{ij} & \text{if } f_{ij} > 0 \\ 0 & \text{if } f_{ij} = 0 \end{array}$	Log	LOGA
$\begin{array}{ll} \frac{1+\log f_{ij}}{1+\log a_j} & \text{if } f_{ij} > 0 \\ 0 & \text{if } f_{ij} = 0 \end{array}$	Normalized log	LOGN
$\begin{array}{ll} 0.5 + 0.5 \left(\frac{f_{ij}}{x_j}\right) & \text{if } f_{ij} > 0 \\ 0 & \text{if } f_{ij} = 0 \end{array}$	Augmented normalized term frequency	ATF1

Table 3: Established local weight formulas used.

The document vectors and query vectors are weighted using separate schemes. The local weight is computed according to the terms in the given document or the query. The global weight, however, is based on the document collection regardless of whether we are weighting documents or queries. The normalization is done after the local and global weighting. Normalizing the query vectors is not necessary because it does not affect the relative order of the ranked document list.

We present well-known weighting schemes in this section and several new schemes in the next section. In Sect. 5, we compare the methods.

Local weighting formulas perform well if they work on the principle that the terms with higher within-document frequency are more pertinent to that document [9]. A list of the established local weight formulas we used is given in Table 3.

The simplest local weights are binary (BNRY) [9] and within-document frequency (FREQ) [9], given respectively by

$$
\begin{aligned}
L_{ij} &= \begin{cases} 1, & \text{if } f_{ij} > 0 \\ 0, & \text{if } f_{ij} = 0 \end{cases}, \quad \text{and} \\
L_{ij} &= f_{ij} \ ,
\end{aligned}
$$

where f_{ij} is the frequency of term i in document j. These weights are typically used for query weighting, where terms appear only once or twice. For document weighting, these weights are generally not best because BNRY does not differentiate between terms that appear frequently and terms that appear only once and because FREQ gives too much weight to terms that appear frequently. The logarithm offers a middle ground.

Logarithms are used to adjust within-document frequency because a term that appears ten times in a document is not necessarily ten times as important as a term that appears once in that document. Two of the local weighting formulas in Table 3 are similar because they each use logarithm. They are log (LOGA) [5] and normalized log (LOGN) [1], given respectively by

$$L_{ij} = \begin{cases} 1+\log f_{ij}, & \text{if } f_{ij} > 0 \\ 0, & \text{if } f_{ij} = 0 \end{cases}, \quad \text{and}$$

$$L_{ij} = \begin{cases} \dfrac{1+\log f_{ij}}{1+\log a_j}, & \text{if } f_{ij} > 0 \\ 0, & \text{if } f_{ij} = 0 \end{cases},$$

where a_j is the average frequency of the terms that appear in document j.[2] Because LOGN is normalized by the $(1 + \log a_j)$ term, the weight given by LOGN will always be lower than the weight given by LOGA for the same term and document. When no global weight is used, it is important to use a normalized local weight. LOGN and LOGA are the favored local document and query weights, respectively, in recent TRECs [1, 2].

Another local weight that is a middle ground between binary and term frequency is augmented normalized term frequency (ATF1) [9]:

$$L_{ij} = \begin{cases} .5 + .5\left(\dfrac{f_{ij}}{x_j}\right), & \text{if } f_{ij} > 0 \\ 0, & \text{if } f_{ij} = 0 \end{cases},$$

where x_j is the maximum frequency of any term in document j. ATF1 awards weight to a term for appearing in the document and then awards additional weight for appearing frequently. With this formula, L_{ij} varies only between 0.5 and 1 for terms that appear in the document.

Global weighting tries to give a "discrimination value" to each term. Many schemes are based on the idea that the less frequently a term appears in the whole collection, the more discriminating it is [9]. A commonly used global weight is the inverted document frequency measure, or IDF, derived by Sparck Jones [11]. We have used two variations, IDFB [9] and IDFP [4], given respectively by

$$G_i = \log\left(\frac{N}{n_i}\right), \quad \text{and}$$

$$G_i = \log\left(\frac{N - n_i}{n_i}\right),$$

where N is the number of documents in the collection and n_i is the number of documents in which term i appears. IDFB is the logarithm of the inverse of the probability that term i appears in a random document. IDFP is the logarithm of the inverse of the odds that term i appears in a random document. IDFB and IDFP are similar in that they both award high weight for

[2] All logs are base two.

Formula	Name	Abbr.
$\log\left(\frac{N}{n_i}\right)$	Inverse document frequency	IDFB
$\log\left(\frac{N-n_i}{n_i}\right)$	Probabilistic inverse	IDFP
$1+\sum_{j=1}^{N}\frac{\frac{f_{ij}}{F_i}\log\frac{f_{ij}}{F_i}}{\log N}$	Entropy	ENPY
$\frac{F_i}{n_i}$	Global frequency IDF	IGFF
1	No global weight	NONE

Table 4: Established global weight formulas used.

terms appearing in few documents in the collection and low weight for terms appearing in many documents in the collection; however, they differ because IDFP actually awards negative weight for terms appearing in more than half of the documents in the collection, and the lowest weight IDFB gives is one.

In addition, we used the Entropy weight (ENPY) [8] given by

$$G_i = 1+\sum_{j=1}^{N}\frac{\frac{f_{ij}}{F_i}\log\frac{f_{ij}}{F_i}}{\log N},$$

where F_i is the frequency of term i throughout the entire collection. If a term appears once in every document, then that term is given a weight of zero. If a term appears once in one document, then that term is given a weight of one. Any other combination of frequencies will yield a weight somewhere between zero and one. Entropy is a useful weight because it gives higher weight for terms that appear fewer times in a small number of documents.

We also used a global frequency IDF weight (IGFF) [5], given by

$$G_i = \frac{F_i}{n_i} \ .$$

Here, if a term appears once in every document or once in one document, it is given a weight of one, the smallest possible weight. A term that is frequent relative to the number of documents in which it appears gets a large weight. This weight often works best when combined with a different global weight on the query vector.

For comparison, we also used no global weight (NONE); that is, the global weight assigned to term i is one. A list of these established global weight formulas is given in Table 4.

Formula	Name	Abbr.
$\frac{1}{\sqrt{\sum_{i=0}^{m}(G_i L_{ij})^2}}$	Cosine normalization	COSN
$\frac{1}{(1 - slope) + slope\ l_j}$	Pivoted unique normalization	PUQN
1	None	NONE

Table 5: Normalization factors used.

The third component of the weighting scheme is the normalization factor, which is used to correct discrepancies in document lengths. It is useful to normalize the document vectors so that documents are retrieved independent of their lengths. See Table 5 for a list of the normalization factors we used.

Perhaps the most familiar form of normalization in the vector space model is cosine normalization (COSN) [9]:

$$N_j = \frac{1}{\sqrt{\sum_{i=0}^{m}(G_i L_{ij})^2}} \ ,$$

which divides by the magnitude of the weighted document vector, thereby forcing the magnitude of the weighted document vectors to be one. This allows us to compare the angle between the weighted vectors.

With COSN, longer documents are given smaller individual term weights, so smaller documents are favored over longer ones in retrieval. Pivoted unique normalization (PUQN) [10], a relatively new normalization method, tries to correct the problem of favoring short documents. PUQN is given by

$$N_j = \frac{1}{(1 - slope)\, pivot + slope\ l_j} \ ,$$

where l_j is the number of distinct terms in document j. Per the suggestion of [10], *slope* is set to 0.2 and *pivot* is set to the average number of distinct terms per document in the entire collection. The basic principle behind pivoted normalization methods is to correct for discrepancies based on document length between the probability that a document is relevant and the probability that the document will be retrieved. Using another normalization factor, such as $1/l_j$, a set of documents is retrieved, and the retrieval and the relevance curves are plotted against document length. The point at which these curves intersect is the *pivot*. The documents on the left side of the pivot generally have a higher probability of being retrieved than they have of being relevant, and the documents on the right side of the pivot generally have a higher probability of being relevant than they have of being retrieved. The normalization factor can now be pivoted at the pivot and tilted so that the normalization factor can be increased or decreased to better match the probabilities of relevance and retrieval [10].

We also used no normalization at all (NONE), where N_j is set to one.

Document weight	Query weight	Scheme name
LOGA ENPY COSN	LOGA ENPY	Log-entropy [5]
LOGA IGFF COSN	ATF1 ENPY	IGFF-entropy
FREQ NONE NONE	FREQ NONE	Raw term frequency [9]
FREQ NONE COSN	FREQ NONE	Raw cosine [9]
LOGA NONE COSN	LOGA IDFB	SMART "ltc" [1]
LOGA NONE COSN	LOGA IDFP	Variation of SMART "ltc"
FREQ IDFB COSN	ATF1 IDFB	Best fully weighted [9]
ATF1 NONE NONE	BNRY IDFP	Best probabilistic [9]
LOGN NONE PUQN	LOGA IDFB	Pivoted unique new norm weight [10]

Table 6: Popular weighting schemes.

A list of popular combinations of the weights we have discussed is given in Table 6.

4. New Term Weight Formulas

We have developed several new local and global weightings. The new local weights are listed in Table 7. Two new local weighting formulas are changed-coefficient ATF1 (ATFC) and augmented average term frequency (ATFA), given respectively by

$$L_{ij} = \begin{cases} 0.2 + 0.8\left(\frac{f_{ij}}{x_j}\right), & \text{if } f_{ij} > 0 \\ 0, & \text{if } f_{ij} = 0 \end{cases}, \quad \text{and}$$

$$L_{ij} = \begin{cases} 0.9 + 0.1\left(\frac{f_{ij}}{a_j}\right), & \text{if } f_{ij} > 0 \\ 0, & \text{if } f_{ij} = 0 \end{cases}.$$

ATFC was developed using a general version of ATF1 found in [3]:

$$L_{ij} = \begin{cases} K + (1-K)\left(\frac{f_{ij}}{x_j}\right), & \text{if } \ f_{ij} > 0 \\ 0, & \text{if } \ f_{ij} = 0 \end{cases}.$$

Changed-coefficient ATF1 works well because, like ATF1, it assigns weight to a term merely for appearing in a document, then adds more weight if the term appears frequently in the document. The difference is in the coefficients. With ATFC, even more weight is given if a term appears frequently in a document but less weight is given just for appearing. ATFA is similar to ATF1, but it is normalized differently. ATF1 is normalized by the maximum within-document frequency of a particular document, and ATFA is normalized by the average within-document frequency of a document. Also, the coefficients are different. ATFA gives more weight to a term for just appearing and adds less weight if a term appears frequently. Note that the maximum value for ATFC is one, whereas one is the average value for ATFA.

Formula	Name	Abbr.
$0.2 + 0.8\left(\frac{f_{ij}}{x_j}\right)$ if $f_{ij} > 0$ 0 if $f_{ij} = 0$	Changed-coefficient ATF1	`ATFC`
$0.9 + 0.1\left(\frac{f_{ij}}{a_j}\right)$ if $f_{ij} > 0$ 0 if $f_{ij} = 0$	Augmented average term frequency	`ATFA`
$0.2 + 0.8\log(f_{ij} + 1)$ if $f_{ij} > 0$ 0 if $f_{ij} = 0$	Augmented log	`LOGG`
$\sqrt{f_{ij} - 0.5} + 1$ if $f_{ij} > 0$ 0 if $f_{ij} = 0$	Square root	`SQRT`

Table 7: New local weight formulas.

Another new local weight is augmented log (LOGG), a variation of ATFC, given by

$$L_{ij} = \begin{cases} 0.2 + 0.8\log(f_{ij} + 1), & \text{if } f_{ij} > 0 \\ 0, & \text{if } f_{ij} = 0 \end{cases}.$$

We simply changed $\left(\frac{f_{ij}}{x_j}\right)$ to $\log(f_{ij} + 1)$ because log seems to be a better local weight than within-document frequency. Note that now L_{ij} can be greater than one.

Our fourth new local weight is square root (SQRT), given by

$$L_{ij} = \begin{cases} \sqrt{f_{ij} - 0.5} + 1, & \text{if } f_{ij} > 0 \\ 0, & \text{if } f_{ij} = 0 \end{cases}.$$

In the development of SQRT we tried to model a formula whose graph resembled that of LOGA, a top performer among established local weight formulas. We looked at the graph of LOGA and noted that the function $\sqrt{f_{ij}}$ would have a similar shape. We then translated the square root curve until it resembled the log curve more closely. As f_{ij} gets large, SQRT has a larger value than LOGA.

We have three new global weights, as shown in Table 8. The first is log-global frequency IDF (IGFL), given by

$$G_i = \log\left(\frac{F_i}{n_i} + 1\right).$$

IGFL is simply a combination of the IDFA and IGFI weights.[3] We noticed that the established IDF weights were all logarithm functions. We also observed that the IGFI weight was working well, so we combined the two formulas.

[3] IGFL was discovered by accident when another IDF weight was incorrectly implemented in the program. Before the mistake was noticed, we found that this IDF weight worked better than any of the other IDF weights. Unfortunately, when we fixed the mistake, that IDF weight was no longer superior. We then implemented the "incorrect" formula in the program, calling it IGFL.

Formula	Name	Abbr.
$\log\left(\frac{F_i}{n_i}+1\right)$	Log-global frequency IDF	IGFL
$\frac{F_i}{n_i}+1$	Incremented global frequency IDF	IGFI
$\sqrt{\frac{F_i}{n_i}-0.9}$	Square root global frequency IDF	IGFS

Table 8: New global weight formulas.

The second new global weight is square root global frequency IDF (IGFS), given by

$$G_i = \sqrt{\frac{F_i}{n_i} - 0.9} \ .$$

Like IGFL, IGFS is a combination of formulas. In this case, we observed that square root was an excellent local weight, so we adapted it to be a global weight. We found that subtracting larger numbers from F_i/n_i improved performance. We do not subtract one because that could cause a global weight of zero for some terms.

The third new global weight is incremented global frequency IDF (IGFI), given by

$$G_i = \frac{F_i}{n_i} + 1 \ .$$

When trying to develop new and improved local weights, it was found that adding one to a formula significantly improved its performance, so we thought it might carry over to the global weights. Since IGFF already performed best, we tried adding one to it, and the result was IGFI.

5. Results

To test these weighting formulas, we implemented the vector space model in C and tested our weighting schemes on several test sets that include the "correct answers." For a given weighting scheme, we computed the similarity between the documents and each query in the test collection and returned a list of documents ranked in order of their similarity scores. We then computed two scores: *interpolated average precision* (IAP) and *Top Ten*. In IAP, the ranked document list is checked to see where the relevant documents placed; the best IAP score, 100, would be given if all the relevant documents were ranked first. (See Kolda [7] for further description of IAP.) Another rating of the performance of weighting schemes is the number of relevant documents in the Top Ten returned documents. We report the average IAP and Top Ten scores over all the queries in each collection.

We tested both the established and new weighting schemes on the MEDLINE, CRANFIELD, and CISI test collections; Tables 9, 10, and 11 show the results. The various weighting schemes are listed from highest to lowest in terms of IAP. Also listed is the average number of relevant documents in the first ten documents retrieved (Top Ten). The new weighting formulas are denoted by an asterisk (*). Note that the performance (both absolute and relative) of the various weighting schemes varies depending on the test collection.

As the results show, our new term weighting formulas offer improvement over the popular weighting schemes. The new weights work well combined with both other new weights and with established weights. The combination of local and global weights used does make a difference. A particular local weight when combined with one global weight may perform well but when combined with a different global weight may perform poorly. The combination of document weighting and query weighting also makes a difference in performance.

Our results show that local weight SQRT works well in both documents and queries. For all three test collections, SQRT is in the best document weighting scheme with respect to IAP. Furthermore, SQRT appears in most of the top-performing weighting schemes. With respect to the Top Ten measure, SQRT always appears in the top five weighting schemes. SQRT works better than its predecessor, LOGA, possibly because SQRT gives more weight for terms appearing only a few times in a given document.

LOGG is another new local weight that performs well. The results tables show that LOGG consistently appears in the top five document weighting schemes for all three test collections with respect to IAP. LOGG also works well as a query local weight. The results show that LOGG appears in weighting schemes that always perform better than the popular ones with respect to Top Ten.

Local weights ATFC and ATFA also perform well. Our results show that these weights are in weighting schemes that perform better than the popular weighting schemes with respect to IAP. With respect to Top Ten, ATFC and ATFA appear in schemes that perform better than the popular weighting schemes.

Our results show that the new global weight IGFS performs well when it is combined with any of the new local weight formulas. IGFS consistently appears in the top weighting schemes with respect to both IAP and Top Ten.

The new global weight IGFI also performs well in combination with any of the new local weight formulas. The results tables show that IGFI consistently appears in the top five weighting schemes with respect to both IAP and Top Ten.

The new global weight IGFL consistently performs better than the popular weighting schemes with respect to IAP and tends to work best with the SQRT and ATFC local weights. IGFL always appears in the top ten weighting schemes with respect to Top Ten.

For each test collection, the best new weighting schemes offered improvement over the best popular weighting schemes in terms of IAP. This is a 3.3% improvement in MEDLINE, 2.8% in CRANFIELD, and 7.0% in CISI.

With respect to Top Ten, the best new weighting schemes also offered improvement over

Document weight	Query weight	IAP	Top 10
SQRT* IGFF COSN	BNRY IDFB	59.55	6.83
SQRT* IGFS* COSN	BNRY IDFP	59.05	6.80
SQRT* IGFS* COSN	BNRY IDFB	59.01	6.83
LOGG* IGFS* COSN	ATFA* ENPY	58.98	6.77
SQRT* IGFI* COSN	BNRY IDFB	58.91	6.87
LOGG* IGFF COSN	SQRT* IDFB	58.70	6.73
ATFA* IGFS* COSN	BNRY IDFB	58.43	6.90
LOGG* IGFS* COSN	LOGG* IDFP	58.33	6.63
ATFC* IGFS* COSN	BNRY IDFP	58.33	6.87
LOGG* IGFI* COSN	SQRT* ENPY	58.31	6.77
ATFA* IGFS* COSN	SQRT* ENPY	58.17	6.70
ATFC* IGFI* COSN	BNRY IDFB	57.87	6.83
SQRT* IGFS* COSN	LOGA IDFP	57.87	6.47
SQRT* IGFI* COSN	ATFC* ENPY	57.82	6.63
ATFC* IGFL* COSN	BNRY IDFB	57.68	6.87
SQRT* IGFL* COSN	LOGG* IDFB	57.67	6.67
LOGA IGFF COSN	ATF1 ENPY	57.67	6.63
LOGA NONE COSN	LOGA IDFP	53.41	6.20
LOGA NONE COSN	LOGA IDFB	53.29	6.17
LOGA ENPY COSN	LOGA ENPY	52.92	6.30
FREQ IDFB COSN	ATF1 IDFB	52.47	6.17
LOGN NONE PUQN	LOGA IDFB	52.44	6.00
ATF1 NONE NONE	BNRY IDFP	51.85	6.27
FREQ NONE COSN	FREQ NONE	48.19	5.67
FREQ NONE NONE	FREQ NONE	42.81	5.20

Table 9: Results on the MEDLINE test collection using various weighting schemes. The results are sorted from highest to lowest by IAP.

Document weight	Query weight	IAP	Top 10
SQRT* IGFL* COSN	LOGG* IDFB	43.06	3.02
SQRT* IGFI* COSN	BNRY IDFB	43.04	3.03
SQRT* IGFI* COSN	ATFC* ENPY	43.00	3.03
LOGG* IGFI* COSN	SQRT* ENPY	42.88	3.03
SQRT* IGFF COSN	BNRY IDFB	42.70	3.04
LOGG* IGFS* COSN	ATFA* ENPY	42.67	3.02
SQRT* IGFS* COSN	BNRY IDFB	42.64	3.00
LOGG* IGFF COSN	SQRT* IDFB	42.53	3.04
SQRT* IGFS* COSN	BNRY IDFP	42.50	3.01
LOGG* IGFS* COSN	LOGG* IDFP	42.36	3.02
SQRT* IGFS* COSN	LOGA IDFP	42.19	3.00
ATFC* IGFI* COSN	BNRY IDFB	42.14	2.92
ATFC* IGFL* COSN	BNRY IDFB	42.02	2.94
ATFA* IGFS* COSN	SQRT* ENPY	41.93	2.95
LOGA IGFF COSN	ATF1 ENPY	41.90	2.92
ATFA* IGFS* COSN	BNRY IDFB	41.79	2.94
ATFC* IGFS* COSN	BNRY IDBP	41.76	2.96
LOGA NONE COSN	LOGA IDFB	41.52	2.90
LOGN NONE PUQN	LOGA IDFB	41.20	2.93
LOGA NONE COSN	LOGA IDFP	41.13	2.88
LOGA ENPY COSN	LOGA ENPY	39.85	2.86
ATF1 NONE NONE	BNRY IDFP	38.92	2.81
FREQ IDFB COSN	ATF1 IDFB	38.52	2.80
FREQ NONE COSN	FREQ NONE	35.71	2.60
FREQ NONE NONE	FREQ NONE	24.55	1.86

Table 10: Results on the CRANFIELD test collection using various weighting schemes. The results are sorted from highest to lowest by IAP.

Document weight	Query weight	IAP	Top 10
SQRT* IGFS* COSN	LOGA IDFP	19.40	2.91
LOGG* IGFS* COSN	LOGG* IDFP	19.21	2.91
SQRT* IGFI* COSN	ATFC* ENPY	18.92	3.00
SQRT* IGFS* COSN	BNRY IDFP	18.90	2.74
SQRT* IGFL* COSN	LOGG* IDFB	18.78	3.00
LOGG* IGFF COSN	SQRT* IDFB	18.65	3.14
ATFA* IGFS* COSN	SQRT* ENPY	18.58	2.94
SQRT* IGFS* COSN	BNRY IDFB	18.56	3.06
SQRT* IGFF COSN	BNRY IDFB	18.50	2.97
ATFA* IGFS* COSN	BNRY IDFB	18.42	2.94
LOGG* IGFS* COSN	ATFA* ENPY	18.38	3.03
ATFC* IGFS* COSN	BNRY IDBP	18.28	2.94
LOGG* IGFI* COSN	SQRT* ENPY	18.22	3.09
SQRT* IGFI* COSN	BNRY IDFB	18.19	2.89
LOGN NONE PUQN	LOGA IDFB	18.13	2.91
LOGA ENPY COSN	LOGA ENPY	18.07	2.89
LOGA NONE COSN	LOGA IDFB	18.03	2.86
LOGA NONE COSN	LOGA IDFP	18.03	2.74
ATFC* IGFL* COSN	BNRY IDFB	17.90	2.86
ATFC* IGFI* COSN	BNRY IDFB	17.89	2.86
FREQ IDFB COSN	ATF1 IDFB	17.67	2.86
LOGA IGFF COSN	ATF1 ENPY	17.64	3.00
ATF1 NONE NONE	BNRY IDFP	17.61	2.40
FREQ NONE COSN	FREQ NONE	15.22	2.00
FREQ NONE NONE	FREQ NONE	13.51	1.91

Table 11: Results on the CISI test collection using various weighting schemes. The results are sorted from highest to lowest by IAP.

the best popular weighting schemes. This improvement is an average of 0.27 more relevant documents retrieved in MEDLINE, 0.11 in CRANFIELD, and 0.14 in CISI.

6. Conclusions

The goal of information retrieval is to make it easy for the user to obtain data relevant to a given query and to do so automatically. The success or failure of the vector space method depends on the term weighting schemes. We have developed new term weighting formulas that improve upon the existing ones.

We found that simpler weighting formulas work best for the query weighting. This is possibly because each term in a query appears only once or twice. For the queries, we recommend local weighting formulas like BNRY and the new LOGG and global weighting formulas such as ENPY or any IDF weight.

We found that more complex term weight formulas are necessary for the documents, possibly because documents contain more terms, these terms occur with greater frequency, and length discrepancies are more noticeable in the documents than in the queries. For the documents, we recommend any of the new local weighting formulas, especially SQRT. The various IGF weights are the best choices for document global weights.

The weighting schemes with the two best average IAP scores are a document weight of `SQRT*-IGFF-COSN` combined with a query weight of `BNRY-IDFB` or a document weight of `SQRT*-IGFS*-COSN` combined with a query weight of `BNRY-IDFP`. We recommend these as the best weighting schemes.

In recent years, improvements have been made on information retrieval systems by using phrases, expansion, and clustering. We are confident that our new formulas will improve the efficiency of these methods as well.

7. References

[1] C. Buckley, A. Singhal, M. Mitra, and G. Salton. New retrieval approaches using SMART: TREC 4. In D. K. Harman, editor, *The Fourth Text REtrieval Conference* (TREC-4), pages 25–48. NIST Special Publication 500-236, Gaithersburg, Maryland, 1996. http://trec.nist.gov/.

[2] C. Buckley, J. Walz, M. Mitra, and C. Cardie. Using clustering and superconcepts within SMART : TREC 6. In E. M. Voorhees and D. K. Harman, editors, *The Sixth Text REtrieval Conference* (TREC-6), pages 107–124. NIST Special Publication 500-240, Gaithersburg, Maryland, 1998. http://trec.nist.gov/.

[3] W. B. Croft. Experiments with representation in a document retrieval system. *Information Technology: Research and Development*, 2:1–21, 1983. Cited in [6].

[4] W. B. Croft and D. J. Harper. Using probabilistic models of document retrieval without relevance information. *J. Documentation*, 35(4):285–295, 1979. Cited in [6].

[5] S. Dumais. Improving the retrieval of information from external sources. *Behavior Research Methods, Instruments, and Computers*, 23:229–236, 1991.

[6] D. Harman. Ranking algorithms. In W. B. Frakes and R. Baeza-Yates, editors, *Information Retrieval Data Structures and Algorithms*, pages 363–392. Prentice Hall, 1992.

[7] T. G. Kolda. *Limited-Memory Matrix Methods with Applications.* PhD thesis, Applied Mathematics Program, University of Maryland, College Park, Maryland, 1997. Also available as Department of Computer Science Technical Report CS-TR-3806.

[8] K. E. Lochbaum and L. Streeter. Comparing and combining the effectiveness of latent semantic indexing and the ordinary vector space model for information retrieval. *Information Processing and Management*, 25(6):665–676, 1989. Cited in [6].

[9] G. Salton and C. Buckley. Term weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523, 1988.

[10] A. Singal, C. Buckley, M. Mitra, and G. Salton. Pivoted document length normalization. Technical Report TR95-1560, Department of Computer Science, Cornell University, Ithaca, New York, 1995.

[11] K. Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *J. Documentation*, 28(1):1–21, 1972.

A. Documents and Query from Tables 1 and 2

Document 1:

Placental and cord blood lipids. Comparison in a set of double ovum twins, a stillborn and a live-born.

1. Determinations of phospholipid, total and free cholesterol, triglyceride and nefa have been made on placental tissue and cord blood in a set of double ovum twins, one stillborn and one live-born.

2. Similarities occurred in all fractions studied except the cord blood triglyceride and nefa levels.

3. The serum of the stillborn infant contained one-third as much triglyceride and 21/2 times as much nefa as did the live-born infant.

4. The phospholipid content and the total lipid content of the stillbirth placenta were the highest studied in this laboratory which includes determinations on 26 live births.

5. The suggestion is made that increased lipoprotein lipase activity in the cord blood may accompany intrauterine fetal death.

Document 2:

Correlation between maternal and fetal plasma levels of glucose and free fatty acids.

Correlation coefficients have been determined between the levels of glucose and ffa in maternal and fetal plasma collected at delivery. significant correlations were obtained between the maternal and fetal glucose levels and the maternal and fetal ffa levels. From the size of the correlation coefficients and the slopes of regression lines it appears that the fetal plasma glucose level at delivery is very strongly dependent upon the maternal level whereas the fetal ffa level at delivery is only slightly dependent upon the maternal level.

Query:

The crossing of fatty acids through the placental barrier. Normal fatty acid levels in placenta and fetus.