MIS Review Vol. 19, No. 1, September (2013), pp. 61-80
DOI: 10.6131/MISR.2013.1901.04

An Effective Pareto Optimality Based Fusion Technique for Information Retrieval

Krishnan Batri
Department of Electronics and Communication Engineering, PSNA College of Engineering and Technology, India

ABSTRACT: *Information Retrieval (IR) is the process of retrieving information that is relevant to the users' needs. Over the years, researchers tend to develop the best retrieval strategy, which achieves the best possible performance across all document collections. Their results indicate a pattern of tug-of-war relationship prevalent among the existing strategies, where in one strategy dominates the remaining strategies over other document collections. Data Fusion may nullify the aforesaid tug-of-war effect. It can extract the best possible performance among the participating members. Data Fusion in IR usually combines the various retrieval schemes (strategies) to enhance the overall system performance. Our proposed fusion functions assign relevance scores by considering non dependency among all participating strategies. Relevance score assignment based on the relationship between that specific document and all other documents in the corpus. The existing Comb functions treated as the baseline functions for our proposed functions. Proposed and baseline functions' performance tested among three medium size corpuses. The average precision value of functions indicates that, one of our proposed functions achieves better performance in comparison with the base line functions. The statistical analysis confirms the same.*

KEYWORDS: *Information Retrieval, Data Fusion, Meta Search, Vector Space Model, Similarity Measures, Extended Boolean Model.*

1. Introduction

Knowledge sharing helps the mankind in the evolution process. Once they start sharing their knowledge, they became more civilized. Knowledge sharing process made up of two critical tasks. The first one deals with the creation and the second one associated with the sharing. Apart from knowledge creation, knowledge sharing seems to be more critical as it indirectly preserves the knowledge. Hence, this process rendering an important help for the welfare of mankind. Information Retrieval (IR) is the process of sharing the knowledge among the needy people. We may call it as science, art, or a technique, but it meticulously store, organize, and proffer the required information.

Information retrieval is the process of retrieving the required information according to the users' needs (Baeze-Yates & Ribeiro-Neto, 1999; Korfhage, 1997; Salton & McGill, 1983). According to Yates, "Information retrieval deals with the representation, storage, organization of and access to the information items (Baeze-Yates & Ribeiro-Neto). The representation and organization of the information items should provide the user with easy access to the information in which he or she is interested."

The information retrieval process made up of three important tasks. The first one deals with the representation of the information. As the available information may be in structured, semi structured and unstructured format, they should be represented in a common format. In order to carry out this task, some preprocessing mechanisms have to be carried out. As the pre-processing mechanisms being the foundation step, variations in their performance alter the overall system's performance. Hence, researchers focused more towards the pre-processing mechanisms and contribute more. Plenty of works carried out on some of the pre-processing mechanisms like, stemming, tokenization, and stop word removal. Based on their results, we came to a final conclusion that the performances of these pre-processing mechanisms are not unique. They tend to vary from one application to other.

Organizing and storing of keywords lies at the middle level. Information should be organized in a manner that the matching process becomes an easier one. This task termed as the indexing process. The indexing process should discriminate the keywords, and store them in a proper manner. The discrimination process involves with the assignment of weights to all keywords. Hence, the indexing process involves with the storing of index terms along with their calculated weights.

The critical nature of the indexing process proves to be more vital than the preprocessing step. Hence, plenty of works carried out in the indexing mechanisms. These works critically divided in to two main categories. The first one deals with the data structure, which used to store the index terms. The second one involves with the calculation of index terms' weights. The weighting mechanism is more critical as it has the ability to alter the overall system's performance. It should discriminate a keyword from the rest. Method of calculating the index term's weight vary from one approach to other. Hence, the performance of the weighting mechanism is not unique. It varies from one application to other.

Method of calculating the correlation between the user's query and the information source is the last task in information retrieval process. Actually the retrieval system finds the relevant information in this last task because the information source available prior to the query, and they are indexed. The query posted at the last minute. The previous two tasks assist the last one. In other words, these three tasks are dependent. They should be

executed in a sequential order. If there is a deviation in any one of the task, it will affect the entire retrieval process.

The corpus may contain one or more relevant information sources. If it has only one source, than there won't be any problem. If there are few or more, than one important question will surface out. Which one is more relevant? Answer to this question demands a measuring method, which used to measure the correlation between the query, and the information sources. It termed as the similarity measure. The principle of operation of a similarity measure depends on the underlying storage model and the weight assignment mechanism. Hence, the literature flooded with different types of similarity measures.

Once the retrieval systems calculate the degree of correlation, the next question will pop up. How to list the relevant sources? Enumeration of the relevant sources should be based on their degree of correlation. Hence, the final task subdivided in to two sub tasks. The first sub task calculates the correlation and the second one list the sources based on their correlation. In some retrieval system, there is no provision for listing. Apart from this, the different retrieval systems use the different weighting schemes. As the performance of the weighting schemes is not consistent, the performance of the retrieval system also varies.

Different methods used to implement the three different retrieval tasks of the retrieval process. As the choices are plenty, there is a need for standardization. Various models used to standardize the retrieval process. The retrieval model expresses the method of processing, organizing, storing, and retrieving the information. Based on their operating principle, the models classified in to three types. They are (1) Exact match model, (2) Vector space model, and (3) Language model. Each of these models has its own advantages, and disadvantages. These models incorporate the three primary tasks of the retrieval processes differently. Hence, the performance of these retrieval models not consistent.

From the above discussions, we come to a conclusion that, performance of retrieval models and the three tasks is not consistent. It is varying. If we propose a better model or mechanism near future, it will also render an inconsistent performance. Even more, our proposed mechanism will lose the battle ground against some other new models or mechanisms. In engineering, there is a scope for improvement. Hence, instead of spending our energy to develop a new model, why can't we tap the positive potential of these existing models and mechanisms. Answer to this leads to the development of new research area called data fusion. The data fusion merges the merits of underlying mechanisms or models. If a better model or mechanism evolves, we can add it to the pool. Hence the data fusion has some scope of enhancement. Hence, the data fusion seems to be better than the individual models or schemes. It proves to be better.

Rest of this article organized as follows. Section 2 gives the details about the retrieval models and data fusion principles. Section 3 gives the insight about the earlier works in the area of information retrieval and data fusion. Section 4 gives the details about our proposed work. Section 5 gives the details about the experimental setup and the results. Section 6 concludes with the future direction of our research.

2. Models of IR and data fusion

This section dedicated to IR models and data fusion. Various types of models and their underlying principles, their comparison are discussed in the first part of this section. The concept of data fusion along with its needs, and its principles are given in the last part of this section.

2.1 IR models

IR models used to describe the principles associated with each, and sub tasks of the retrieval process. More specifically a *model* is a set of premises and an algorithm for ranking documents with regard to a user query (Salton & McGill, 1983). More formally, an IR model is a quadruple [D, Q, F, $R(q_i, d_j)$] where D and Q is a set of logical views of documents and queries and $R(q_i, d_j)$ is a ranking function which associates a numeric ranking to the query q_i and the document d_j and F is the frame work for modelling document and queries. *Strategy* or scheme is synonymous with rank $R(q_i, d_j)$. It is a method of assigning similarity between the query and the documents. *System* refers to the physical implementation of an IR algorithm which can have various operational modes or various settings of parameters. Therefore the same IR system may be used to execute different IR schemes by adjusting the various parameters.

Performance of the IR system depends on the underlying IR algorithm, which in turn depends on the underlying IR model. IR models classified in to three types. Out of which, exact match model is most primitive. Lots of works carried out on the vector space model. It is almost saturated. Language models are still in developing state. Hence, works are going on in the language model. Hence, we focus more towards the first two models. In future, we plan to accommodate the language model.

2.1.1 Exact match model

Boolean model is very simple, and it operates on the principle of Boolean algebra. It retrieves documents based on a word matching function. Since the decision space in Boolean Model is binary, documents judged as either relevant or irrelevant. Thus the number of documents retrieved as a result of the Boolean nature of the model is either vast or too small. Also there is no provision for the ranking the documents. These limitations

eliminated by extending the Boolean Model with the functionality of partial matching and term weighting. This extended model combines the advantages of the Boolean model and the VSM.

Salton introduced the Extended Boolean Model (EBM) on 1983. In this model, the weights assigned to the terms lie between zero to one. It uses the maximum normalization method, and the normalized weights assigned to the index terms. The function used in maximum normalization given in Equation (1).

$$normalized\ w_{i,j} = \frac{unnormalized\ w_{i,j}}{w_l} \tag{1}$$

Where,

$w_{i,j}$ = weight of the term i in j^{th} document and

w_l = maximum weight of the generic index term l in the corpus.

The weight assignment techniques in the EBM are same as that of VSM with the only difference being that the weights are normalized. The matching function or similarity measure adapted from the Boolean Model. In Extended Boolean Model, a query represented in one of the following forms: (1) Conjunctive form, (2) Disjunctive form, and (3) Combination of both conjunctive and disjunctive form.

In disjunctive query form, distance from (0,0) used as the similarity measure between the query and the document. Conjunctive query form uses (1,1) as the origin for the distance measure. The distance measure not restricted to Euclidean distance but generalized to any value ranging from 1 to ∞. As EBM depends on the value of p (distance) for calculating the similarity value, it also referred as P-norm model. The generalized form of the query in conjunctive and disjunctive form is represented in Equations (2) and (3) respectively.

$$q_{or} = (w_1 \vee^p, w_2 \vee^p, w_3 \vee^p, \cdots\cdots\cdots, w_m \vee^p) \tag{2}$$

$$q_{and} = (w_1 \wedge^p, w_2 \wedge^p, w_3 \wedge^p, \cdots\cdots\cdots, w_m \wedge^p) \tag{3}$$

The similarity measure between the document, and the query in the P-norm model given in Equations (4) and (5).

$$Sim(q_{or}, d_j) = \left(\frac{w_1^p + w_2^p + w_3^p + \cdots + w_m^p}{m}\right)^{1/p} \tag{4}$$

$$Sim(q_{and}, d_j) = 1 - \left(\frac{(1-w_1)^p + (1-w_2)^p + \cdots + (1-w_m)^p}{m}\right)^{1/p} \tag{5}$$

Where,

w_m = weight of the index term and $1 \leq p \leq \infty$

2.1.2 Vector space model

Vector Space Model (VSM) is the most popular IR model. VSM not only explains the process of retrieving relevant documents but also the assignment of rank to the documents. In VSM, the objects of IR, such as term, document, and query treated as multidimensional linearly dependent vectors in the vector space.

In vector space model, the weight $w_{t,d}$ associated with the index term "t" in a document "d" is positive and non binary. Furthermore, the index terms in the query also weighted. Let $w_{t,q}$ be the weight associated with the pair [t,q], where $w_{t,q} \geq 0$. Then the query vector q is defined in Equation (6).

$$q = (w_{1,q}, w_{2,q}, w_{3,q}, w_{l,q}, \ldots, w_{n,q}) \tag{6}$$

Where,

n = the total number of index terms in the system,

$w_{l,q}$ = weight of the index term l in query q.

In VSM, the documents represented as a linear combination of keywords or index terms. The weights of the index terms can be calculated in many ways. Since the objects of IR treated as being linearly dependent, term vector can be represented as a linear combination of documents. The term vector defined as follows:

$$t_l = (d_1, d_2, d_3, \ldots, d_N), \quad 1 \leq l \leq n$$

Where,

d_N = weight of the t_l^{th} term in the d_N^{th} document,

N = total number of documents in the corpus and

n = total number of index terms in the corpus.

The scalar product between the query, and document vectors used to calculate the relevance (similarity) of the document with respect to the query. In the vector space V, if there exist two vectors x and y such that $x, y \in V$ then the scalar product defined in Equation (7).

$$S(x, y) = |x| \cdot |y| \cdot \cos\theta \tag{7}$$

Where,

|x|, |y| = magnitude of the vectors,

$| x | = \sqrt{\sum_{i=1}^{n} x_i^2}$,

$| y | = \sqrt{\sum_{i=1}^{n} y_i^2}$ and,

θ = angle between two vectors.

The vector space that considers only the scalar product termed as the Euclidean space. The scalar product used as one of the methods to calculate the correlation between the query, and the document vectors. There are various methods available to calculate the correlation (similarity) value. Based on the value of the correlation between the query and the document, the relevance of the document justified. The retrieved documents arranged in descending order based on the value of their similarity.

2.2 Data fusion

Effectiveness of the existing IR system depends on the underlying model and strategy. Certain strategies perform well in a specific environment while their performance deteriorates in other environments. Early research shows that there is no single strategy that achieves constant performance across all test document collections (Zobel & Moffat, 1988). As a result of the ever increasing users of Internet, and hence the massive information repository, there is a challenging research requirement to work out a strategy whose effectiveness should be high compared to the other existing retrieval strategies.

Recent research work identifies that fusion technique improves and stabilize the IR system performance (Fox & Shaw, 1994; 1995; Lee, 1995; 1997a; 1997b). Fusion is the methodology of combining retrieval strategies associated with the retrieval task followed by an assignment of relevance score or rank to documents on the basis of the score returned by the fused strategies (Bartell, Cottrell & Belew, 1994; Belkin, Kantor, Cool & Quatrain, 1994; Vogt, 1999). Fusion methods broadly classified into: (1) Data fusion, and (2) Collection fusion. The detailed classification of fusion techniques is shown in Figure 1.

The data fusion approaches combine the results obtained from various retrieval strategies over the same document collection or corpus, whereas collection fusion combines the results of various document collections. In Collection fusion, the same query operates over the various document collections. The relevant documents returned from the multiple corpuses merged together to give the final relevant documents list. In IR, same document, and query can be represented using different weighting scheme. If the

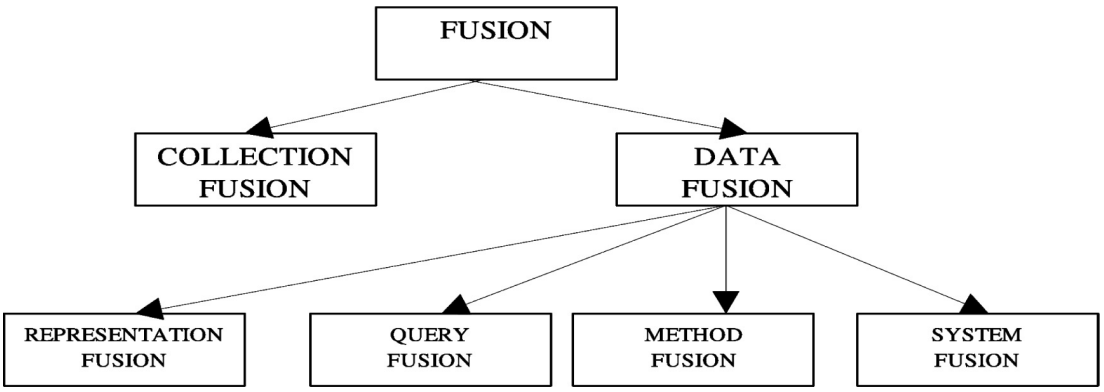

Figure 1 Types of Fusion Techniques

fusion operation merges the result of the various document representations then it termed as representation fusion. If the various query forms fused then it termed as query fusion. Various methods used to retrieve relevant document, and if these methods fused together then it termed as method fusion. The results from the multiple systems merged then it is termed as system fusion. Previous results show that, the fusion methods render some positive impact over the effectiveness of the retrieval system. It also yields consistent results over all test document collections.

3. Prior work in fusion technique

This section details on the various research studies that have been carried out on *Meta Search Algorithm for Combining Scores*. Researchers have exploited the characteristics of "*Meta Search Algorithm for Combining Scores*" by fusing the various existing IR models. Meta Search Algorithm combines the relevance scores returned from the various retrieval strategies to identify the relevant documents.

Early work on data fusion method that does not use training data commenced in the early 70's when Fisher (Fisher & Elchesen, 1972) fused two Boolean searches together. In his method, one search operates on the title word while the other search operates on manually generated index terms. He achieved significant improvement in effectiveness of the information retrieval system. This method combines minimum number of retrieval strategies whereas linear combination method successfully combines more number of strategies. The linear combination method assigns weights to the individual strategies (Vogt, 2000; Vogt & Cottrell, 1999). The final relevance score of a document assigned by weighted linear combination method is given in Equation (8).

$$R(q, d) = \sum_{i=1}^{k} \theta_i \cdot E_i(q, d) \qquad (8)$$

Where,

θ_i = Weight of the i^{th} retrieval strategy,

E_i (q,d) = Relevance score returned by the i^{th} retrieval strategy and

k = Number of retrieval strategies to be fused.

The weighted linear combination method has the limitation of requiring prior knowledge about the retrieval systems to assign the weights. This limitation is eliminated in Comb-functions by treating all strategies equally.

The Comb functions for combining scores by treating all strategies equally have been proposed by Fox and Shaw (Fox & Shaw, 1994). The various Comb Functions used for combining scores is shown in Figure 2.

Fusion techniques differ from Comb-functions in the fact that the relevance is computed on the basis of rank assigned to documents as compared to the relevance scores methodology adopted in Comb-functions. Few such fusion techniques that emulate the social voting schemes are Boarda fusion and Condorcet fusion (Montague & Aslam, 2002). Extensive work on Comb functions has been carried out by Lee. New rationales and indicators for data fusion have been proposed by Lee. He has conducted experiments over TREC data collection. He concluded that CombMNZ is the better performing function than the other Comb-functions.

The training data involved in the fusion techniques are used to assign weights to individual strategies. The weighted scores from the individual strategies are combined linearly to assign the final relevance score. The weighted linear combination method maintains the same weight for all retrieval systems. But the performance of the system differs from query to query. Hence the selection of the best performing retrieval strategy becomes vital. Probabilistic approach is used for this purpose. The best performing strategy is selected automatically from the pool. The probabilistic model selects only one strategy from the pool and all other strategies become idle. Hence evolutionary algorithms are used to select the best performing strategies (Coello, 2000). Billhardt, Borrajo and Maojo (2003) proposed a heuristic based data fusion algorithm. They uses Genetic

CombMIN	*Minimum of Individual Similarities*
CombMAX	*Maximum of Individual Similarities*
CombSUM	*Summation of Individual Similarities*
CombANZ	*CombSUM ÷ Number of Nonzero Similarities*
CombMNZ	*CombSUM × Number of Nonzero Similarities*

Figure 2 Comb-functions for Combining Scores

algorithm to combine the retrieval score. Their algorithm not only assigns the scores to independent strategies but also selects the best performing strategy for fusion.

Fusion techniques utilize the advantages of its member strategies by combining the strategies together. By combining the strategies it tends to exploit the following effects as indicated by Vogt (1999): (1) Skimming effect, (2) Chorus effect and (3) Dark Horse effect. The *Skimming Effect* happens when retrieval approaches that represent their collection items differently may retrieve different relevant items, so that a combination method that takes the top ranked items from each of the retrieval approaches will push non-relevant items down in the ranking. The *Chorus Effect* occurs when several retrieval approaches suggest that an item is relevant to a query; this tends to be a stronger evidence for relevance than that of a single approach. The *Dark Horse Effect* is one in which a retrieval approach may produce unusually accurate (or inaccurate) estimates of relevance for at least some items, relative to the other retrieval approaches. By carefully designing the combination function, researchers utilized the advantages of these above said effects. Our proposed function employs the advantages of skimming effect.

The performance of the fusion techniques that requires training data depends on the relevance feedback from the active participant. So the performance of such systems differs from user to user and depends on their skill of predicting relevance of the documents. Hence we concentrate on complete user independent fusion techniques. Literature survey has established Comb-functions to be the best performing functions in this category. In this work, we compare the performance of our proposed functions over the CombMNZ function, the best among the Comb-functions.

4. Proposed work

This section discusses about our *Pareto Optimality* based fusion technique and its comparison with (1) the existing CombMNZ, a best Meta search algorithm for combining scores and (2) remaining Comb functions detailed in the previous section. A Meta search algorithm combines the results obtained from more than one data source. In IR, fusion algorithm operates on various retrieval strategies to calculate the final relevance score of the document. These retrieval strategies act as the criteria for selection of relevant document. The Decision Vector for Multi Criteria Selection in IR is represented mathematically as in Equation (9).

$$s_i(q, d) = \{s_i^1(q, d), s_i^2(q, d), \ldots\ldots, s_i^j(q, d)\} \tag{9}$$

Where,

i = document index

j = number of retrieval strategies to be fused

s_i^j (q, d) = relevance score returned by the j^{th} retrieval strategy.

In multi criteria selection, similarity values of a document from various retrieval strategies are treated as an independent variable of a decision vector. Hence the final decision about the relevance of the document cannot be arrived at by operating only on the vector space. Relevance of the document is decided by calculating the equivalent scalar value of the decision vector. We use the notion of *Pareto Optimality* to calculate the equivalent scalar value (FRS). According to Pareto optimality, in a maximization problem, a vector $s_i^* \in V$ is said to be Pareto optimal "if all other vectors have smaller value for at least one retrieval strategy or have the same value for all retrieval strategies." In other words

s_i^* is said to be Pareto Optimal, iff $\forall j \; s_i^j = s_k^j$ *or*

at least one value of l such that $l \in j, \; s_i^j > s_k^l$

There are various methods available to calculate the final scalar value. Our proposed approach treats all retrieval strategies equally. In order to maintain equality we normalize the relevance scores and use maximum normalization function. The mathematical equation to calculate the normalized score under maximum normalization is represented in Equation (10).

$$s_{normalized} = \frac{s_{unnormalized}}{s_{\max}} \tag{10}$$

Where,

$s_{unnormalized}$ = relevance score returned by a retrieval strategy and

s_{max} = maximum relevance score returned by a generic retrieval strategy.

In our proposed method of combining relevance scores, assignment of final relevance score to a document is based on the relationship between the corresponding document and all other remaining documents in the corpus. We choose the difference between the scores with respect to each of the document for the same retrieval strategy as a metric to establish a relationship. The relevance score difference between documents obtained via all retrieval strategies are of two types namely: (1) Minimum and (2) Maximum difference. The relationship between two documents based on the above mentioned relevance score difference is expressed mathematically as in Equations (11) and (12).

$$d_i \cong X + d_j \quad j = 1, 2, 3, \ldots\ldots N,\ j \neq i \tag{11}$$

$$d_i \cong Y + d_j \quad j = 1, 2, 3, \ldots\ldots N,\ j \neq i \tag{12}$$

Where,

d_i = a specific document to be compared with the other remaining document,

X = minimum difference between two documents in all retrieval strategies,

Y = maximum difference between two documents in all retrieval strategies and

N = total number of documents in the corpus.

For an example, take the relevance score returned by the four retrieval strategies for document 1 and 2 as

d_1 = {5,4,3,2}

d_2 = {1,2,3,1}

Now calculate the difference between the document 1 and 2

$d_1 - d_2$ = {4,2,0,1}

Now take the maximum and minimum value from the above calculated difference

Maximum difference (Y) = 4

Minimum difference (X) = 0

Now the relationship between the document 1 and 2 is

$d_1 \cong 4 + d_2$

$d_1 = 0 + d_2$

Based on the value of minimum and maximum difference, we establish the relationship for a specific document with all other remaining documents in the corpus. The relationship space R of a specific document consists of (N-1) minimum and (N-1) maximum differences. From the relationship space R, we find out either the maximum or minimum value that is globally optimal. We choose one of the global minimum or maximum. The mathematical representation of global maximum and global minimum is given in Equations (13) and (14).

$$x_i \prec x_j \quad iff \quad \forall j(x_i < x_j), i \neq j,\ j = 1, 2, \ldots (N-1). \tag{13}$$

$$x_i \prec x_j \quad iff \quad \forall j(x_i > x_j), i \neq j,\ j = 1, 2, \ldots (N-1). \tag{14}$$

Based on the local and global relationships, we derive four functions to assign the final relevance score to the documents. The proposed functions are (1) C-maxmax, (2) C-maxmin, (3) C-minmax and (4) C-minmin. The formulas used to assign final relevance score based on C functions are given in Figure 3.

One of the main advantages of the proposed method is that it does not require any weight assignment and training data. Since our proposed approach treats all strategies equally, our proposed approach exploits the advantages of "*skimming effect*."

5. Experimental results

This section details on the experimental results of our proposed functions which were discussed in the previous section. We conducted the experiments on three test document collections, namely, (1) CRANFIELD, (2) CISI and (3) ADI under an uniform environment. The document abstracts in CRAN collection are about Aeronautics and these documents are compiled by cranfield institute of technology. The CISI dataset is about library science and it is collected by Institute of Scientific Information. ADI is a small data set in the field of Information Science. Table 1 shows the characteristics of the three

$$FRS_{C-\max\max} = \max_{\forall j, j\neq k}\left(\max_{i=1,2..n}\left(s_i^k - s_i^j\right)\right)$$

$$FRS_{C-\max\min} = \max_{\forall j, j\neq k}\left(\min_{i=1,2..n}\left(s_i^k - s_i^j\right)\right)$$

$$FRS_{C-\min\max} = \min_{\forall j, j\neq k}\left(\max_{i=1,2..n}\left(s_i^k - s_i^j\right)\right)$$

$$FRS_{C-\min\min} = \min_{\forall j, j\neq k}\left(\min_{i=1,2..n}\left(s_i^k - s_i^j\right)\right)$$

Where,
j,k = Document id,
i = retrieval strategy
s_i^j = relevance score of j^{th} document in i^{th} retrieval strategy

Figure 3 C-functions for Combining Scores

Table 1 Characteristics of Datasets

Characteristics	ADI	CISI	MED
Number of documents	82	1,460	1,033
Number of terms	374	5,743	5,831
Number of queries	35	35	30
Average number of document relevant to a query	5	8	23
Average number of terms per document	45	56	50
Average number of terms per query	5	8	10

datasets. We measured the 11 point interpolated precision to judge the performance of our retrieval strategy.

5.1 Retrieval strategies

Retrieval Strategy is used to assign similarity score between the document and the query. We use various similarity measures of VSM and P-norm model as retrieval strategies and fuse them. Various similarity measures of VSM used in our experiment are given in Equations (15) ~ (18).

$$\text{Cosine Similarity } R(q,d) = \frac{\sum_{i \in q \cap d} w_{q,t} \cdot w_{d,t}}{W_q \cdot W_d} \tag{15}$$

$$\text{Inner Product } R(q,d) = \sum_{i \in q \cap d} w_{q,t} \cdot w_{d,t} \tag{16}$$

$$\text{Dice Coefficient } R(q,d) = \frac{\sum_{i \in q \cap d} w_{q,t} \cdot w_{d,t}}{W_q^2 + W_d^2} \tag{17}$$

$$\text{Jaccard } R(q,d) = \frac{\sum_{i \in q \cap d} w_{q,t} \cdot w_{d,t}}{W_q^2 + W_d^2 - \sum_{i \in q \cap d} w_{q,t} \cdot w_{d,t}} \tag{18}$$

Where,

R: Relevance score of document d with respect to query q,

$w_{q,d}$: weight of the term t in the query q,

$w_{d,t}$: weight of the term t in the document d,

W_q: weight of the query and

W_d: weight of the document d.

The conjunctive query form of P-norm model mentioned in Equation (5) is also used as a retrieval strategy in our experiment. We use p value as 1.5, 2.5 and 3.5 to calculate the similarity score in P-norm model. We use the above seven retrieval strategies to test the effectiveness of our proposed functions over (1) CombMNZ, the best meta-search algorithm used for fusion and (2) remaining Comb functions.

We maintain a uniform environment by using the same stop-word list, stemmer algorithm and weight assignment mechanism. The formulas used to assign weight to index terms are given in Equations (19) ~ (21). We use Term Frequency-Inverse Document Frequency (TF-IDF) weight assignment schemes for assigning weights to index terms.

$$w_t = \log_{10}\left(1+\frac{N}{f_t}\right) \tag{19}$$

$$w_{d,t} = r_{d,t} \cdot w_t \tag{20}$$

$$r_{d,t} = f_{d,t} \tag{21}$$

Where,

w_t = term weight

$w_{d,t}$ = document term weight

$r_{d,t}$ = relative term frequency

$f_{d,t}$ = frequency of the term t in document d

5.2 Results

We conducted the experiments by combining the various similarity measures of VSM along with the P-norm similarity measures. We chose P value as 1.5, 2.5, and 3.5. We used 11 point interpolated precision to calculate the effectiveness of the proposed functions. We also used the average value of 11-point interpolated precision to compare the effectiveness of our proposed functions against the Comb functions. The results for the proposed C-functions and the Comb functions are shown in Figure 4.

The averages of 11 point interpolated precision are given in Table 2 to make the comparison process easy. The last column shows the average of all functions over all document collections. The table indicates that C-minmax is the best performing function compared to remaining C functions and Comb functions; it achieves 5.95% improvement over the CombMNZ function.

The CombMNZ function is the subset of linear combination model. The linear combination model assigns weights to the retrieval strategies exploiting the *Chorus effect*. CombMNZ function treats all strategies equally and assigns equal weights to all

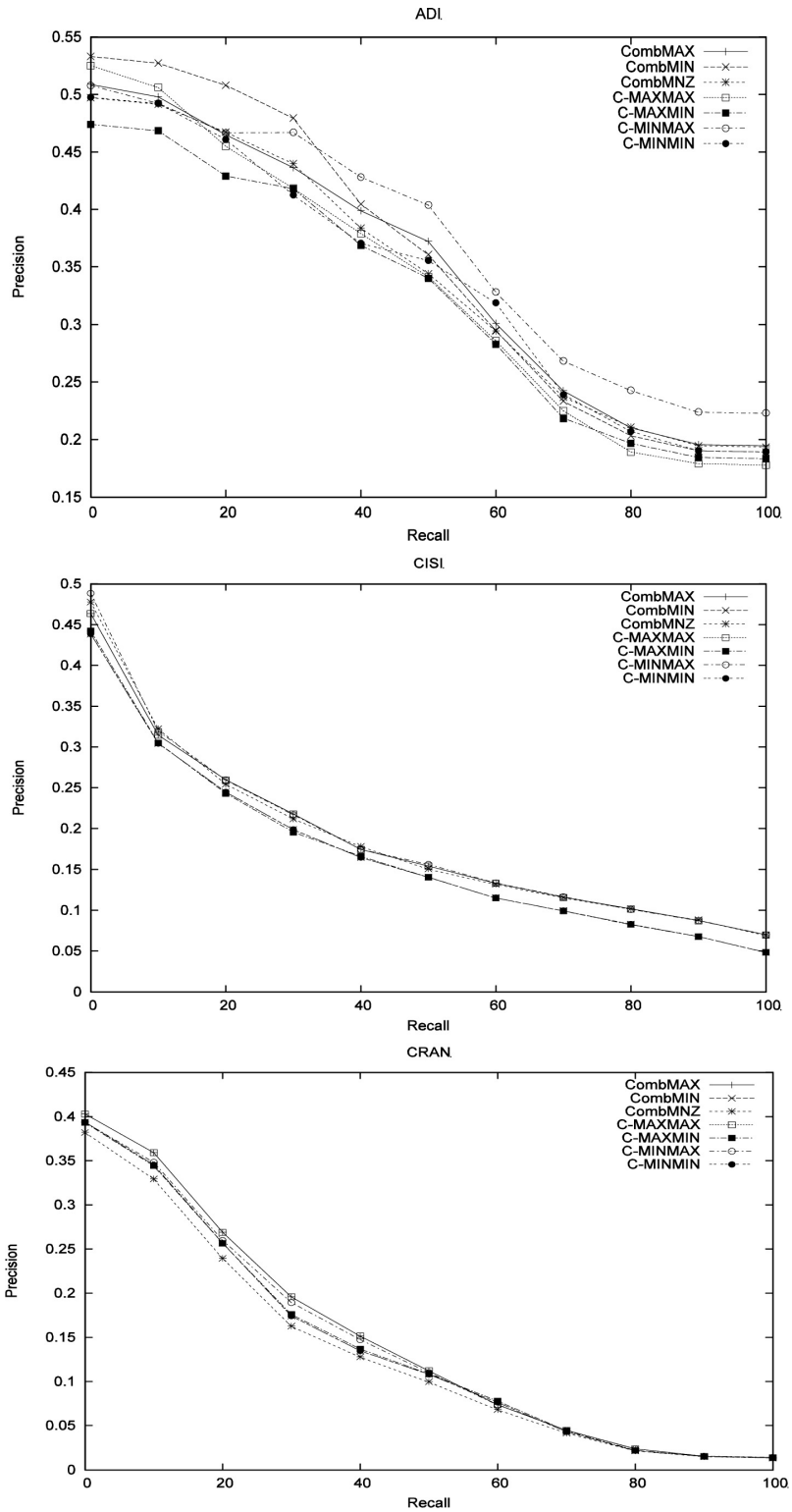

Figure 4 11-pt Interpolated Precision Curve for C and Comb Functions

Table 2 Average 11-pt Interpolated Precision for C and Comb Functions

Function	ADI	CISI	CRAN	Average
CombMNZ	0.3413	0.1911	0.1364	0.2229
CombMAX	0.3475	0.1901	0.1511	0.2296
CombMIN	0.3569	0.1733	0.1439	0.2247
C-Maxmax	0.3347	0.1901	0.1511	0.2253
C-Maxmin	0.3240	0.1732	0.1444	0.2139
C-Minmax	0.3685	0.1929	0.1473	0.2362
C-Minmin	0.3395	0.1733	0.1439	0.2189

retrieval strategies. Since CombMNZ assigns equal weights to all strategies and it is one of the subsets of linear combination model, it exploits the advantages of both chorus and skimming effects. Our proposed C-functions incorporate skimming effect, since it treats all strategies equally. The proposed approach does not combine the scores linearly and the final relevance score depends on the individual scores as compared to the existing linear combination of scores methodology adopted in CombMNZ. Also the proposed approach is based only on skimming effect whereas CombMNZ utilizes both skimming and chorus effects. The limitation of such a strategy is that a large Chorus effect cuts into the possible gain from the skimming effect, thereby leading to degradation in performance.

In our proposed functions of C-maxmax and C-minmax, the right half that is $\max_{i=1,2..n}\left(s_i^k - s_i^j\right)$ part maximizes the difference between the similarity scores returned from the corresponding retrieval strategies. The ideal relevance score of a relevant document is set as one and for the non relevant it is set as zero. Hence the maximum allowed difference value is one. Since the sub portion $\max_{i=1,2..n}\left(s_i^k - s_i^j\right)$ maximizes the difference, it indirectly chooses the document which has the relevance score more close to the ideal value. As a result both of the C-maxmax and C-minmax functions perform well in our experiment. But the slight degradation in performance of C-maxmax function is due to the Dark horse effect. Few retrieval strategies unexpectedly give maximum scores to non relevant documents. C-maxmax chooses the maximum value from the calculated difference as compared to C-minmax which chooses the minimum difference. Hence C-minmax becomes the best performing function compared to the remaining C-functions and Comb functions.

6. Conclusion

We have proposed a set of new functions for combining multiple relevance scores in information retrieval. The proposed functions do not require any training data and return only the relevance scores as compared to ranks being returned by other existing fusion methods. The average value of the 11 point interpolated precision over the three test document collections shows that the C-minmax is the better performing function compared to remaining C-functions and Comb functions. The proposed functions treat all strategies equally like that of Combfunctions. The proposed approach compute the relationship between the documents, hence it require some extra computation time. The C-functions introduced in this paper is very much useful for medium size document collection.

References

Baeze-Yates, R. and Ribeiro-Neto, B. (1999), *Modern Information Retrieval*, Addison Wesley, Harlow, UK.

Bartell, B.T., Cottrell, G.W. and Belew, R.K. (1994), 'Automatic combination of multiple ranked retrieval systems', *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Dublin, Ireland, July.

Belkin, N.J., Kantor, P., Cool, C. and Quatrain, R. (1994), 'Combining evidence for information retrieval', *Proceedings of the Second Text REtieval Conference*, Gaithersburg, MD, August/September.

Billhardt, H., Borrajo, D. and Maojo, V. (2003), 'Learning retrieval expert combinations with genetic algorithm', *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, Vol. 11, No. 1, pp. 87-113.

Coello, C.A. (2000), 'An updated survey of GA-based multiobjective optimization techniques', *ACM Computing Survey*, Vol. 32, No. 2, pp. 109-143.

Fisher, H.L. and Elchesen, D.R. (1972), 'Effectiveness of combining title words and index terms in machine retrieval searches', *Nature*, Vol. 238, pp. 109-110.

Fox, E.A. and Shaw, J.A. (1994), 'Combination of multiple searches', *Proceedings of the Second Text REtieval Conference*, Gaithersburg, MD, August/September.

Fox, E.A. and Shaw, J.A. (1995), 'Combination of multiple searches', *Proceedings of the Third Text REtieval Conference*, Gaithersburg, MD, November.

Korfhage, R.R. (1997), *Information Storage and Retrieval*, John Wiley & Sons, New York, NY.

Lee, J.H. (1995), 'Combining multiple evidence from different properties of weighting schemes', *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Seattle, WA, July.

Lee, J.H. (1997a), 'Analyses of multiple evidence combination', *Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Philadelphia, PA, July.

Lee, J.H. (1997b), 'Combining multiple evidence from different relevant feedback networks', *Proceedings of the 5th International Conference on Database Systems for Advanced Applications*, Melbourne, Australia, April.

Montague, M.H. and Aslam, J.A. (2002), 'Condorcet fusion for improved retrieval', *Proceedings of the 11th International Conference on Information and Knowledge Management*, McLean, VA, November.

Salton, G. and McGill, M.J. (1983), *Introduction to Modern Information Retrieval*, McGraw-Gill, New York, NY.

Vogt, C.C. (1999), 'Adaptive combination of evidence for information retrieval', Unpublished doctoral dissertation, University of California, San Diego, CA.

Vogt, C.C. (2000), 'How much more is better? Characterizing the effects of adding more IR systems to a combination', *Proceedings of RIAO 2000: 6th International Conference on Content-Assisted Information Retrieval*, Paris, France, April.

Vogt, C.C. and Cottrell, G.W. (1999), 'Fusion via a linear combination of scores', *Information Retrieval*, Vol. 3, No. 1, pp. 151-173.

Zobel, J. and Moffat, A. (1988), 'Exploring the similarity space', *ACM SIGIR Forum*, Vol. 1, No. 32, pp. 18-34.

About the author

Krishnan Batri is Professor at Department of Electronics and Communication Engineering, PSNA College of Engineering and Technology, Dindigul. He has completed hid Ph.D from National Institute of Technlogy, Trichy on 2008. His area of interest includes Information Retrieval, Text Mining, and Genetic Algorithm. His research papers published in various international journal and conferences.

Corresponding author. Department of Electronics and Communication Engineering, PSNA College of Engineering and Technology, Muthanampatty, Dindigul, TamilNadu, India. Tel: +91-9789680969. E-mail address: krishnan.batri@gmail.com