

FUSION METHODS BASED ON COMMON ORDER INVARIABILITY FOR META SEARCH ENGINE SYSTEMS

Xiaohua Yang

Department of Computer Science

South-China University
Hengyang, Hunan 421001
China

xhyang@ctit.zhnut.edu.cn

Hui Yang Minjie Zhang

School of Information Technology &
Computer Science

University of Wollongong
NSW 2522
Australia

hy92,minjie@uow.edu.au

Abstract: Fusion is an important work for running meta search engine systems. A fusion algorithm is rational iff it can satisfy three conditions, i.e. the associativity, commutativity and common order invariability. Among these conditions, the associativity and commutativity can be satisfied by all existing fusion methods, but the common order invariability is only adhered by parts of fusion methods.

In this paper, we present a general fusion method, which is based on the common order invariability, for meta search engine systems. This method is essentially a framework that can combine with any other fusion method to form a rational fusion method. Therefore, it provides a simple but effective way to transform an unreasonable rank fusion method into a rational one.

Keywords: Internet, Information Retrieval, Meta Search Engine, Rank, and Fusion

INTRODUCTION

In general, a search engine is essentially a ranking Internet information retrieval system. It accepts users' queries, calculates scores of relevance between indexed documents and queries, ranks indexed documents against the relevant scores and selects top ranked indexed documents to constitute results. Altavista, Excite, Google, Hotbot, Infoseek and Lycos etc. are examples of currently popular Internet search engines.

A meta search engine is a special Internet information retrieval agent on the top of other search engines [W. S. Beuermann, 1998]. Queries are submitted to the meta search engine, which in turn sends the query to multiple back-end search engines (in parallel) and then merges multiple results offered by different search engines. Metasearch, SavvySearch, Metacrawler, Profusion and Inquirus etc. are examples of meta search engines.

One of the main technical problems of running a meta search engine is the fusion of multiple results of back-end search engines. Fusion problem has been studied in distributed information retrieval systems for many years. In [J. P. Callan, 1995], four typical rank fusion methods are introduced and compared by J. P. Callan etc. They are interleaving, raw score, normalized score and weighted score. If only the document rankings are available, the results from each collection can be interleaved (interleaving fusion). If the relevant scores of documents are available and the scores from different collections are comparable, multiple results can be merged based on documents' scores directly (raw score fusion). If the scores are incomparable, one can normalize them by standardizing statistics such as *idf* (inverted document frequency) for the set of collections being searched in some cases (normalized

score fusion). An alternative to both simple interleaving and normalized score is merging based on weighted scores. It ranks documents against the product of documents' scores and weights of collections (weighted score fusion).

Many practical meta search engine systems adopt rank fusion methods which are based on the above four methods. MetaCrawler [E. W. Selberg, 1997] uses a confidence score to determine how close a reference matches a query. It first distributes the confidence scores returned by each engine into the range 0 to 1000. Thus, the top pick from each engine will have a confidence score of 1000. Then, MetaCrawler eliminates duplicates, and adds the removed reference's score to the sum of the duplicated references confidence scores. This method is obviously an extension of interleaving (confidence score interleaving). The rank fusion of Profusion [S. Gauch, 1996] is an integration of normalized score and weighted score merging. It consists of three steps. First, Profusion maps the original relevant scores into $[0, 1]$. Then, the normalized scores are multiplied by the estimated accuracy of search engines. Finally, if there are duplicated documents, the maximum of all duplicates' scores will be the final relevant scores of the documents, which will be used to decide rankings of documents. SavvySearch's rank fusion method is a simple extension of normalized score method [A. E. Howe, 1997]. The sum of all duplicates' scores is used to rank the final result.

Kirsch [S. T. Kirsch, 1997] has presented another different typical technique for relevance ranking with meta search engines wherein the underlying search engines are modified to return extra information such as the number of occurrences of each search term in the documents and the number of occurrences in the entire database. The ranking and relevance scores are computed based on this information at the client independently.

The client relevance ranking fusion strategy has been adopted by some meta search engine systems. For example, in order to rank the final result, Inquirus [S. Lorence, 1998] really fetches selected documents and orders them against their relevant scores that are calculated according to the number of query items presenting in documents, the proximity between query terms, terms frequencies and other factors. MetaCrawler [E. W. Selberg, 1997] also provide a user-enabled alternative rank fusion method which download the pages referred by each service and analyze them directly.

As we can see, there are many fusion methods for meta search engines. In [X. Yang, 2000], we have presented necessary constraints for fusion methods to be rational. In this paper, we focus on the construction of rational fusion methods for meta search engines. In section 2, the constraints for fusion algorithms to be rational are introduced and applied to practical fusion methods. Then, a kind of fusion algorithms based on the constraints is presented in section 3. Finally, a brief conclusion is given in section 4.

NECESSARY CONSTRAINTS OF RATIONAL FUSION METHODS

Definition 1. A search engine (SE) is a 4-tuple, $SE = \langle D, Q, r, t \rangle$, where D is the index database of Internet documents, Q is the set of valid queries, r is the rank algorithm and t ($0 < t$) is the threshold for result selection.

Definition 2. A meta search engine (MSE) is a 5-tuple, $MSE = \langle E_m, Q_m, H_m, r_m, t_m \rangle$, where E_m is the set of underlying search engines, Q_m is the set of valid queries, H_m is the set of query transformations, r_m is the rank algorithm and t_m ($0 < t_m$) is the threshold for result selection.

$\forall q \in Q_m$, given by a user, let the relevant document set of original result offered by SE_i ($1 \leq i \leq n$) is $\{d_i^1, d_i^2, \dots, d_i^{l(i)}\}$ ($1 \leq i \leq n$) and the relevant document set of final result produced by MSE is $\{d^1, d^2, \dots, d^l\}$, then the following conditions are true.

- (1) $\{d^1, d^2, \dots, d^l\} \subseteq \cup_{i=1}^n \{d_i^1, d_i^2, \dots, d_i^{l(i)}\}$,
- (2) $\forall k$ ($1 \leq k \leq l$), $t_m \leq r_m(d^k, q)$, and
- (3) $\forall k, \forall j$, if $1 \leq O_m(d^k) \leq O_m(d^j) \leq l$, then $r_m(d^j, q) \leq r_m(d^k, q)$; where $O_m(x)$ is the ranking of x in the ordered list of final result and $r_m(x, q)$ is the relevance score of x .

The original result offered by SE_i can be associated with a vector $R_i(q) = (w_i^1, w_i^2, \dots, w_i^l)$, where $w_i^j = r_i(d^j, q)$ if $d^j \in \{d_i^1, d_i^2, \dots, d_i^{l(i)}\}$ and w_i^j is unknown if $d^j \notin \{d_i^1, d_i^2, \dots, d_i^{l(i)}\}$. And the final result of MSE can also be associated with a vector $R_m(q) = (w_m^1, w_m^2, \dots, w_m^l)$, where $w_m^j = r_m(d^j, q)$. It is commonly supposed that the original documents' rankings or relevant scores are always available, but the rank algorithm of MSE, i.e. r_m , is usually implicit. So, it is the task of fusion to construct the vector $R_m(q)$ from vectors $R_1(q), \dots, R_n(q)$. Since some elements of $R_i(q)$ ($1 \leq i \leq n$) are unknown and the relevant scores of documents given by different search engines are rare comparable directly, the fusion process usually consists of two steps. The first step is to trim each original vector $R_i(q)$ into an uniform formatted vector $\nabla R_i(q) = (\nabla w_i^1, \nabla w_i^2, \dots, \nabla w_i^l)$, where $\nabla: \mathbf{R}^+ \cup \{0, \text{UNKNOWN}\} \rightarrow \mathbf{R}^+ \cup \{0\}$ is a trimming map and \mathbf{R}^+ is the set of positive reals. The second step is to combine multiple vectors $\nabla R_i(q)$ into a single one $R_m(q) = \nabla R_1(q) \oplus \nabla R_2(q) \oplus \dots \oplus \nabla R_n(q)$, where $\oplus: (\mathbf{R}^+ \cup \{0\}) \times (\mathbf{R}^+ \cup \{0\}) \rightarrow \mathbf{R}^+ \cup \{0\}$ is a merging map. In general, we can get the following equation:

$$r_m(d^j, q) = \nabla r_1(d^j, q) \oplus \nabla r_2(d^j, q) \oplus \dots \oplus \nabla r_n(d^j, q).$$

Definition 3. Suppose $MSE = \langle E_m, Q_m, H_m, r_m, t_m \rangle$ is a meta search engine with member search engines $SE_i = \langle D_i, Q_i, r_i, t_i \rangle$ ($i=1, 2, \dots, n$), the fusion algorithm of MSE is a function $f_{\nabla, \oplus}: \mathbf{R}^n \rightarrow \mathbf{R}$, where \mathbf{R} is the set of rank algorithms, satisfying the condition:

$\forall q \in Q_m, r_m(d^j, q) = f_{\nabla, \oplus}(r_1(d^j, q), \dots, r_n(d^j, q)) = \nabla r_1(d^j, q) \oplus \nabla r_2(d^j, q) \oplus \dots \oplus \nabla r_n(d^j, q)$, where d^j is a relevant document of the final result, $\nabla: \mathbf{R}^+ \cup \{0, \text{UNKNOWN}\} \rightarrow \mathbf{R}^+ \cup \{0\}$ is called as the trimming map of $f_{\nabla, \oplus}$ and $\oplus: \mathbf{R}^+ \cup \{0\} \rightarrow \mathbf{R}^+ \cup \{0\}$ is called as the merging map of $f_{\nabla, \oplus}$.

Example 1: (1) In Metasearch's confident score interleaving fusion method, ∇ is the confidence score calculating algorithm and \oplus is just the sum function.

(2) In Profusion's rank fusion method, ∇ consists of two steps: normalizing score and weighting score, and \oplus is the maximum function.

(3) In SavvySearch's rank fusion method, ∇ is the score normalizing algorithm and \oplus is the sum function.

(4) In Inquirus's rank fusion method, ∇ is the client relevance computing algorithm and \oplus is the maximum function.

A rational fusion method must satisfy some necessary constraints [X. Yang, 2000]. First, given a query, the final result of a meta search engine must be only based on the original results returned by underlying search engines, but not on the order of rank fusion. That means

that a rational fusion method must satisfy the following constraints.

(1) **Associativity:**

$$\forall r_i, r_j, r_k \in R, (\nabla r_i \oplus \nabla r_j) \oplus \nabla r_k = \nabla r_i \oplus (\nabla r_j \oplus \nabla r_k).$$

(2) **Commutativity:**

$$\forall r_i, r_j \in R, \nabla r_i \oplus \nabla r_j = \nabla r_j \oplus \nabla r_i.$$

Second, a meta search engine is running on the top of its member search engines. It usually doesn't have its own index database. Even if a meta search engine has an independent document collection, it is usually much smaller than those of underlying search engines. Therefore, while ranking documents in the final result, a meta search engine must respect the original order relationships among documents provided by member search engines. More exactly, if an order relationship is consistent in all original results, it must keep invariant after re-ranked by the meta search engine. This is called the common order invariability.

Suppose that $SE_i = \langle D_i, Q_i, r_i, t_i \rangle$ and $SE_j = \langle D_j, Q_j, r_j, t_j \rangle$ are two member search engines of a meta search engine $MSE = \langle E_m, Q_m, H_m, r_m, t_m \rangle$, q is a given query, $R_i(q)$ and $R_j(q)$ are relevant document sets of original results returned by SE_i and SE_j , respectively, $R(q) = R_i(q) \cup R_j(q)$ is the set of relevant documents of merged result. The common order invariability can be formally described as follows.

(3) **Common order invariability:**

(a) $\forall x \in R_i(q) \cap R_j(q), y \in R_i(q) \cap R_j(q)$, if $r_i(x, q) \leq r_i(y, q)$ and $r_j(x, q) \leq r_j(y, q)$, then $r_m(x, q) \leq r_m(y, q)$.

(b) $\forall x \in R_j(q) - R_i(q), y \in R_i(q) \cap R_j(q)$, if $r_j(x, q) \leq r_j(y, q)$, then $r_m(x, q) \leq r_m(y, q)$.

(c) $\forall x \in R_i(q) - R_j(q), y \in R_i(q) \cap R_j(q)$, if $r_i(x, q) \leq r_i(y, q)$, then $r_m(x, q) \leq r_m(y, q)$.

If we assume that unknown is less than any number, it is easy to have the following conclusion.

Theorem 1. Suppose that $f_{\nabla, \oplus}(r_1, r_2, \dots, r_n) = \nabla r_1 \oplus \nabla r_2 \oplus \dots \oplus \nabla r_n$ is the fusion method of a MSE, if ∇ and \oplus are both monotonic increasing functions, then $f_{\nabla, \oplus}$ can satisfy the common order invariability.

Example 2: According to theorem 1, Metasearch's confident score interleaving fusion method, Profusion's rank fusion method and SavvySearch's rank fusion method can satisfy the common order invariability. Furthermore, they can also satisfy associativity and commutativity. So they are rational rank fusion methods.

Example 3: For a client relevance ranking fusion method, since the final documents' rankings are independent with their original rankings, it cannot satisfy the common order invariability. Therefore, neither Kirsch's fusion algorithm, nor Inquiyus' fusion method, nor MetaCrawler's user-enabled alternative fusion strategy is a rational fusion method.

FUSION METHODS BASED ON COMMON ORDER INVARIABILITY

Let $O_i(x)$ and $O_j(x)$ are ranking positions of x in the original ordered lists of results offered by SE_i and SE_j , respectively, and $O_m(x)$ is the ranking position of x in the fused ordered list, the

common order invariability can be depicted in the following form.

(a') $\forall x \in R_i(q) \cap R_j(q), y \in R_i(q) \cap R_j(q)$, if $O_i(x) \geq O_i(y)$ and $O_j(x) \geq O_j(y)$, then $O_m(x) \geq O_m(y)$.

(b') $\forall x \in R_j(q) - R_i(q), y \in R_i(q) \cap R_j(q)$, if $O_j(x) \geq O_j(y)$, then $O_m(x) \geq O_m(y)$.

(c') $\forall x \in R_i(q) - R_j(q), y \in R_i(q) \cap R_j(q)$, if $O_i(x) \geq O_i(y)$, then $O_m(x) \geq O_m(y)$.

And the general rational fusion algorithms based on common order invariability are presented as follows. (Labels in the left column are used as indicators.)

- (1) $S = R(q)$;
- (2) WHILE ($S \neq \Phi$) DO
- (3) { Select an element x from S ;
- (4) $P(x)=[1, N]$; /* $[1, N]=\{ k \mid 1 \leq k \leq N \}$, N is the size of set $R(q)$ and $P(x)$ is the set of possible ranking positions of document x in the merged ordered list */
- (5) IF $x \in R_i(q) - R_j(q)$ THEN
 $P(x)=[l_x, N]$, where $l_x = 1 + | \{ y \mid y \in R_i(q) \cap R_j(q) \text{ and } O_i(y) < O_i(x) \} |$;
- (6) IF $x \in R_j(q) - R_i(q)$ THEN
 $P(x)=[l_x, N]$, where $l_x = 1 + | \{ y \mid y \in R_i(q) \cap R_j(q) \text{ and } O_j(y) < O_j(x) \} |$;
- (7) IF $x \in R_i(q) \cap R_j(q)$ THEN
 $P(x)=[l_x, u_x]$, where $l_x = 1 + | \{ y \mid y \in R_i(q) \cap R_j(q), O_i(y) < O_i(x) \text{ and } O_j(y) < O_j(x) \} |$,
 $u_x = N - u_1 - u_2 - u_3$, $u_1 = | \{ y \mid y \in R_i(q) - R_j(q) \text{ and } O_i(x) < O_i(y) \} |$, $u_2 = | \{ y \mid y \in R_j(q) - R_i(q) \text{ and } O_j(x) < O_j(y) \} |$, $u_3 = | \{ y \mid y \in R_i(q) \cap R_j(q), O_i(x) < O_i(y) \text{ and } O_j(x) < O_j(y) \} |$;
- (8) $S = S - \{x\}$;
- (9) }
- (10) $S = R(q)$;
- (11) FOR $i=1$ TO N
- (12) { $S_i = \{x \mid x \in S \text{ and } P(x)=[i, u_x] \}$;
- (13) $k = \text{MIN} \{ u_x \mid x \in S_i \}$;
- (14) $T_i = \{x \mid x \in S_i \text{ and } u_x = k \}$;
- (15) Select an element x_i from T_i
- (16) $O_m(x_i) = i$;
- (17) $S_i = S_i - \{x_i\}$;
- (18) WHILE ($S_i \neq \Phi$) DO
- (19) { Select an element x from S_i ;
- (20) $P(x) = P(x) - \{i\}$;
- (21) $S_i = S_i - \{x\}$;
- (22) }
- (23) $S = S - \{x_i\}$;
- (24) }

Lemma 1: At the statement (14) of above program, for any $x \in T_i$, $P(x) = \bigcap_{z \in S_i} P(z)$.

Theorem 2: The above fusion method can satisfy the common order invariability.

Proof:

(1) $\forall x \in R_i(q) \cap R_j(q), y \in R_i(q) \cap R_j(q)$, let $P(x)=[l_x, u_x]$ and $P(y)=[l_y, u_y]$, if $O_i(x) > O_i(y)$ and $O_j(x) > O_j(y)$,

(a) $\forall z \in R_i(q) \cap R_j(q)$, if $O_i(y) > O_i(z)$ and $O_j(y) > O_j(z)$, then $O_i(x) > O_i(z)$ and $O_j(x) > O_j(z)$.
According to statement (7), $I_x > I_y$.

(b) $\forall z \in R(q) = R_i(q) \cup R_j(q)$,

if $z \in R_i(q) \cap R_j(q)$, $O_i(z) > O_i(x)$ and $O_j(z) > O_j(x)$, then $O_i(z) > O_i(y)$ and $O_j(z) > O_j(y)$;

if $z \in R_i(q) - R_j(q)$ and $O_i(z) > O_i(x)$, then $O_i(z) > O_i(y)$;

if $z \in R_j(q) - R_i(q)$ and $O_j(z) > O_j(x)$, then $O_j(z) > O_j(y)$;

According to statement (7), $u_x > u_y$. So, $P(y) \subset P(x)$. According to lemma 1, it is easy to see that $O_m(x) > O_m(y)$.

(2) $\forall x \in R_j(q) - R_i(q)$, $y \in R_i(q) \cap R_j(q)$, it can be proven similarly that if $O_j(x) > O_j(y)$, then $O_m(x) > O_m(y)$.

(4) $\forall x \in R_i(q) - R_j(q)$, $y \in R_i(q) \cap R_j(q)$, it can also be proven similarly if $O_i(x) > O_i(y)$, then $O_m(x) > O_m(y)$.

At statement (15) of the above algorithm, we can use any fusion method to choose the document x_i . In this sense, we have got a general frame that can combine with any available fusion method. It is easy to see that the combined method can always satisfy the common order invariability. In the other hand, if the fusion method used at statement (15) can satisfy the associativity and commutativity, the combined method can also satisfy these constraints.

CONCLUSION

The associativity, commutativity and common order invariability are general constraints that a rational fusion algorithm must satisfy. The associativity and commutativity are satisfied by all existing fusion methods, but the common order invariability is satisfied only by parts of fusion methods.

In this paper, we present a general fusion method, which is based on the common order invariability, for meta search engine systems. This method is essentially a framework that can combine with any other fusion method to form a new rational fusion method. Therefore, it provides a simple but effective way to transform an unreasonable fusion method into a rational one.

REFERENCES:

[W. S. Beuermann, 1998] W. S. Beuermann and M. Schomburg, Internet Information Retrieval? The Further Development of Meta-Searchengine Technology, *In Proceedings of Internet Summit*, Internet Society, 1998.
<http://www.uni-hannover.de/inet98/paper.html>

[J. P. Callan, 1995] J. P. Callan, Z. Lu and W. B. Croft, Searching Distributed Collections With Inference Networks,
In Proceedings of the 18th International Conference on Research and Development in Information Retrieval (ACM SIGIR-95), pp.21-28, 1995.

[S. Gauch, 1996] S. Gauch, G. Wang and M. Gomez, Profusion: Intelligent Fusion from Multiple, Distributed Search Engines, *Journal of Universal Computer Science*, Vol. 9, No. 2, pp.637-649, 1996.

[A. E. Howe, 1997] A. E. Howe and D. Dreilinger, SavvySearch: A Meta-Search Engine that

Learns which Search Engine to Query, *ACM Transactions on Information Systems*, **Vol. 15**, No. 3, pp.195-222, 1997.

[S. T. Kirsch, 1997] S. T. Kirsch. Document retrieval over networks wherein ranking and relevance scores are computed at the client for multiple database documents, *United States Patent #5,659,732*, 1997.

[S. Lorence, 1998] S. Lorence and C. L. Giles. Inquirus, the NECI Meta Search Engine, *Computer Networks and ISDN Systems*, **Vol. 30**, pp.95-105, 1998.

[E. W. Selberg, 1997] E. W. Selberg and O. Etzioni, The MetaCrawler Architecture for Resource Aggregation on the Web, *IEEE Expert*, **Vol. 12**, No. 1, pp.8-14, 1997.

[X. Yang, 2000] X. Yang and M. Zhang, Necessary Constraints for Fusion Algorithms in Meta Search Engine Systems, *In Proceedings of The International Conference on Intelligent Technologies*, pp.409-416, 2000.