Democratic Data Fusion for Information Retrieval Mediators

Yannis Tzitzikas
Department of Computer Science, University of Crete, Greece
Institute of Computer Science, ICS-FORTH
tzitzik@ics.forth.gr

Abstract

Our research presented in this paper concerns the problem of fusing the results returned by the underlying systems to a mediating retrieval system, also called meta-retrieval system, meta-search engine, or mediator. We propose a fusion technique which is based solely on the actual results returned by each system for each query. The final (fused) ordering of documents is derived by aggregating the orderings of each system in a democratic manner. In addition, the fused ordering is accompanied by a level of democracy (alternatively construed as the level of confidence).

Our method does not require any prior knowledge about the underlying systems, therefore it is appropriate for environments where the underlying systems are heterogeneous and autonomous, thus our method is appropriate for the web.

1 Introduction

We consider a domain (collection) D of documents and a set of information retrieval (IR) systems S over that domain. The only constraint that we impose on the set S, is that each system accepts the same set of queries, i.e. bag of words, natural language queries, boolean expressions of terms, etc.

Our research concerns secondary IR systems, also called meta-retrieval systems, or meta-search engines (see [1] for a brief introduction). Roughly, a secondary IR system is a system which is operationally dependent on the functionality provided by other IR systems. Specifically, we consider secondary systems which upon reception of a user query, they forward the query to each underlying system. Subsequently, they retrieve and fuse (merge) the orderings returned by each system.

In this paper, we focus on the problem of fusing the results coming from the underlying systems. We propose a fusion algorithm which is based solely on the actual results returned by each system for each query. The final (fused) ordering is derived by aggregating the orderings of each system in a democratic manner, in particular, we view the fusion problem as an election where the documents correspond to the candidates, the systems correspond to the electors and each ordering corresponds to a voting ticket. In addition, the fused ordering is accompanied by a level of democracy expressing the degree of homophony/antiphony and this factor can be construed by the user as the level of confidence of the answer returned.

An important characteristic of our method is that it does not rely on any prior knowledge about the underlying systems. This means that it is independent from the indexing and the matching methods employed by the underlying systems, thus we can have mediators over information retrieval systems that employ the boolean model, the vector space model [9], the probabilistic model [8], the inference network model [11], the belief network model [7], or any other information retrieval model. This characteristic makes our technique applicable to evolving environments in which there are systems whose functionality evolves unpredictably over time. Moreover our method does not introduce time or bandwidth costs, nor it requires from the underlying systems to support any special communication protocol.

2 Formulating the Problem

Let $D = \{d_1, ..., d_n\}$ be a collection of n documents. Let $S = \{S_1, ..., S_k\}$ be a set of k information retrieval systems over the collection D. Let S_0 be a mediator over the set of systems S. We will also denote this mediator by $[S_1, ..., S_k]$. Upon receiving a query q the mediator S_0, forwards q to each system in S.

When a system S_i receives a query q, it returns a linear ordering of the set D. We will denote this ordering by $(D)_i(q)$, or just by $(D)_i$, if the query q is clear from context. The mediator S_0 after retrieving the orderings returned by the underlying systems, will have to "fuse" them in order to derive a single ordering, denoted by $(D)_0$. The problem that we study in this paper is precisely how to do this fusing.

3 Fusing in a Democratic Manner

Assume a mediator $[S_1, ..., S_k]$ who has forwarded a query q to each of his underlying systems, and let $\{(D)_1, ..., (D)_k\}$ be the returned orderings of D. We proceed to the following definitions:

Def 3.1 Let $(D)_i$ be an ordering of D and let d be a document in D. We denote by $r_i(d)$ the *position*, from the left, of d in $(D)_i$. $\diamond$

For instance, if $(D)_i =< d_1, d_2 >$ then $r_i(d_1) = 1$ and $r_i(d_2) = 2$.

Def 3.2 Let S be a mediator over D and let d be a document in D. The *votes of d over S*, denoted by $V_S(d)$, is given by:

$$V_S(d) = \sum_{S_i \in S} r_i(d)$$

$\diamond$

That is, V_S is a function ($V_S : D \rightarrow Int$) which for each document d returns the sum of the positions of d in the orderings returned by the systems in S. Clearly if S consists of only one system, eg the system S_1, then $V_S(d) = V_{\{S_1\}}(d) = r_1(d)$ for each document d. When the set S will be clear from context, we will denote the function V_S, by just V.

We view the fusion problem as an election, where the documents correspond to the candidates, the systems correspond to the electors, and each ordering $(D)_i$ corresponds to the voting ticket of S_i. More specifically, $r_i(d_j)$ is the vote of S_i for the document d_j. We assume that the "small" votes (numbers) are given to the preferred documents, while the "big" votes are given to the non-preferred. From this perspective, the quantity $V_S(d_j)$ is the sum of the votes that d_j took by the systems in S. This means that the document d which has the smallest $V_S(d)$, is the winner of the elections. Thus we derive the ordering $(D)_0$, by ordering the documents in decreasing order with respect to V_S.

For implementing the democratic data fusion, the mediator needs a data structure of the form:

DOCUMENTS(DOC:URL, V:Int)

Initially, this table is empty, unless the domain D is fixed and a-priori known to the mediator. In this case this table may by construction contain a row for each document of D and the cells of the column V will have the value 0. Upon reception of an ordering $(D)_i$ the mediator executes the following statement:

For each d_j in $(D)_i$ do
$d_j.V := d_j.V + r_i(d_j)$

Some examples are given below:

- Example (A)

$$\begin{aligned} (D)_1 &= < d_1, d_2 > \\ (D)_2 &= < d_2, d_1 > \\ V(d_1) &= 1 + 2 = 3 \\ V(d_2) &= 2 + 1 = 3 \end{aligned}$$

Observe that the documents d_1, d_2 took the same votes. This implies that we cannot derive a single linear ordering of D. Thus we can write: $(D)_0 = \{d_1, d_2\}$. We will return to the equally-voted documents in section 5.

- Example (B)

$$\begin{aligned} (D)_1 &= < d_1, d_2, d_3 > \\ (D)_2 &= < d_1, d_2, d_3 > \\ (D)_3 &= < d_1, d_2, d_3 > \\ V(d_1) &= 1 + 1 + 1 = 3 \\ V(d_2) &= 2 + 2 + 2 = 6 \\ V(d_3) &= 3 + 3 + 3 = 9 \\ (D)_0 &= < d_1, d_2, d_3 > \end{aligned}$$

Here each system returned the same ordering of D. In such cases we say that we have *homophony*. Clearly, if we have homophony we can always derive a single linear ordering of D.

- Example (C)

$$\begin{aligned} (D)_1 &= < d_1, d_2, d_3 > \\ (D)_2 &= < d_1, d_2, d_3 > \\ (D)_3 &= < d_1, d_3, d_2 > \\ V(d_1) &= 1 + 1 + 1 = 3 \\ V(d_2) &= 2 + 2 + 3 = 7 \\ V(d_3) &= 3 + 3 + 2 = 8 \\ (D)_0 &= < d_1, d_2, d_3 > \end{aligned}$$

Here although we have not homophony, we are able to derive a single linear ordering of D.

- Example (D)

$$\begin{aligned}(D)_1 &= < d_1, d_2, d_3 > \\ (D)_2 &= < d_1, d_3, d_2 > \\ (D)_3 &= < d_3, d_1, d_2 > \\ V(d_1) &= 1+1+2=4 \\ V(d_2) &= 2+3+3=9 \\ V(d_3) &= 3+2+1=6 \\ (D)_0 &= < d_1, d_3, d_2 >\end{aligned}$$

- Example (E)

$$\begin{aligned}(D)_1 &= < d_1, d_2, d_3 > \\ (D)_2 &= < d_1, d_3, d_2 > \\ (D)_3 &= < d_2, d_1, d_3 > \\ (D)_4 &= < d_2, d_3, d_1 > \\ (D)_5 &= < d_3, d_1, d_2 > \\ (D)_6 &= < d_3, d_2, d_1 > \\ V(d_1) &= 1+1+2+3+2+3=12 \\ V(d_2) &= 2+3+1+1+3+2=12 \\ V(d_3) &= 3+2+3+2+1+1=12 \\ (D)_0 &= \{d_1, d_2, d_3\}\end{aligned}$$

Here the mediator received the set of all possible orderings of D. In such cases, we say that we have *antiphony*. Clearly, if we have antiphony then each document takes the same votes thus we cannot derive a single linear ordering of D.

4 Computing the Distance of two Orderings

In order to derive the democratic level of the final ordering, we first define the "distance" between two orderings.

Def 4.1 Let D_a, D_b be two linear orderings of D. The *distance* between D_a and D_b, denoted as $dist(D_a, D_b)$, is defined as:

$$dist((D)_a, (D)_b) = \sum_{i=1}^{|D|} |r_a(d_i) - r_b(d_i)| \qquad (1)$$

◇

The distances between the orderings of the examples of section 3 are shown in the next table.

(A)	$dist((D)_1,(D)_2) = 1+1=2$
(B)	$dist((D)_1,(D)_2) = 0$ $dist((D)_1,(D)_3) = 0$ $dist((D)_2,(D)_3) = 0$
(C)	$dist((D)_1,(D)_2) = 0$ $dist((D)_1,(D)_3) = 1+1=2$ $dist((D)_2,(D)_3) = 1+1=2$
(D)	$dist((D)_1,(D)_2) = 1+1=2$ $dist((D)_1,(D)_3) = 1+1+2=4$ $dist((D)_2,(D)_3) = 1+1=2$
(E)	$dist((D)_1,(D)_2) = 1+1=2$ $dist((D)_1,(D)_3) = 1+1=2$ $dist((D)_1,(D)_4) = 2+1+1=4$ $dist((D)_1,(D)_5) = 1+1+2=4$ $dist((D)_1,(D)_6) = 2+2=4$ $dist((D)_2,(D)_3) = 1+1+2=4$ $dist((D)_2,(D)_4) = 2+2=4$ $dist((D)_2,(D)_5) = 1+1=2$ $dist((D)_2,(D)_6) = 2+1+1=4$ $dist((D)_3,(D)_4) = 1+1=2$ $dist((D)_3,(D)_5) = 2+2=4$ $dist((D)_3,(D)_6) = 1+1+2=4$ $dist((D)_4,(D)_5) = 2+1+1=4$ $dist((D)_4,(D)_6) = 1+1=2$ $dist((D)_5,(D)_6) = 1+1=2$

An important question is whether the function *dist* is a *metric function* [5]. Recall that given a non-empty set X , a distance function d on X, is called a *metric* for X , if it is a function which assigns to each pair of points a real number ($d : X \times X \to R$), and it satisfies the following properties for all $x, y, z \in X$:

i $d(x, y) \geq 0$

ii $d(x, y) = 0$ if and only if $x = y$

iii $d(x, y) = d(y, x)$

iv $d(x, y) \leq d(x, z) + d(y, z)$, (the triangle inequality)

In our case, the set X is the of all linear orderings of the set D. Below we prove that the function *dist* is indeed a metric for X.

Proposition 4.1 The function of Def. 4.1 is a metric function.

Proof:

Clearly, the function *dist* satisfies the properties [i][ii][iii]. Below we prove that property [iv] is also satisfied by *dist*.

Let A, B, C be orderings of D. We have

$$dist(A, C) = \sum_{i=1}^{|D|} |r_A(d_i) - r_C(d_i)|$$

$$
\begin{aligned}
dist(A,B) &= \sum_{i=1}^{|D|} |r_A(d_i) - r_B(d_i)| \\
dist(B,C) &= \sum_{i=1}^{|D|} |r_B(d_i) - r_C(d_i)|
\end{aligned}
$$

Since all $r_A(d_i), r_B(d_i), r_C(d_i)$ are integers (actually positive integers), for any $d \in D$ it holds:

$$|r_A(d) - r_C(d)| \leq |r_A(d) - r_B(d)| + |r_B(d) - r_C(d)|$$

since the function $|x-y|$ is a metric for the set of integers. This implies that $dist(A,C) \leq dist(A,B) + dist(B,C)$.

We conclude that the function *dist* is a metric for the set of all orderings of D.

$\diamond$

Having defined the distance between two orderings of D, we can now compute the distance between the ordering returned by each system S_i, and the final (fused) ordering, $(D)_0$, that is $dist((D)_0, (D)_i)$. These distances for the examples of section 3 are shown in the table that follows . Recall that in the examples (A),(E) we have equally voted documents. Thus, for computing the distances for the example (A) we assume that $(D)_0 =< d_1, d_2 >$, while for computing the distances for the example (E) we assume that $(D)_0 =< d_1, d_2, d_3 >$. We will come back to this issue in section 5.

(A)	$dist((D)_0,(D)_1) = 0$ $dist((D)_0,(D)_2) = 2$
(B)	$dist((D)_0,(D)_1) = 0$ $dist((D)_0,(D)_2) = 0$ $dist((D)_0,(D)_3) = 0$
(C)	$dist((D)_0,(D)_1) = 0$ $dist((D)_0,(D)_2) = 0$ $dist((D)_0,(D)_3) = 2$
(D)	$dist((D)_0,(D)_1) = 2$ $dist((D)_0,(D)_2) = 0$ $dist((D)_0,(D)_3) = 2$
(E)	$dist((D)_0,(D)_1) = 0$ $dist((D)_0,(D)_2) = 2$ $dist((D)_0,(D)_3) = 2$ $dist((D)_0,(D)_4) = 4$ $dist((D)_0,(D)_5) = 4$ $dist((D)_0,(D)_6) = 4$

5 Handling Equally-voted Documents

The summation of votes may result to equally voted documents, as it was demonstrated in the examples (A) and (E). In these cases we cannot derive a single linear ordering. However we have to derive a single linear ordering if we want to present the fused ordering to the user, or, and this is the more important, if we want to compute the distances between the fused ordering and the constituent orderings.

Choosing randomly one of the orderings and considering it as the fused one, results to problems when computing distances. For instance, if in example (E) we select randomly one ordering as the fused one, then the distances between $(D)_0$ and $(D)_i$'s range from 0 to the worst case (here 54), but obviously these distances do not reflect the reality.

In order to overcome this problem, we can assume that all equally voted documents reside on the *same* position in the final ordering. In example (E) this means that $r_0(d_1) = r_0(d_2) = r_0(d_3) = 1$, and we can write $(D)_0 =< \{d_1, d_2, d_3\} >$. With this change the distance between $(D)_0$ and any $(D)_i$ equals to $1 + 2... + n - 1$. For instance in the Example (E) these distances equal to 3.

As another example consider a case where: $V(d_1) = 2$, $V(d_2) = 5$, $V(d_3) = 5$, and $V(d_4) = 7$, The fused ordering is $(D)_0 =< d_1, \{d_2, d_3\}, d_4 >$.

Notice that now the final ordering, $(D)_0$, is not always a linear ordering of D, but it can be a partial ordering of D. However this implies that if an underlying system S_i is a mediator, then its response $(D)_i$ can be a partial ordering of D too. For that reason, we reformulate the fusion problem, initially defined in section 2, by allowing the underlying systems to return partial orderings of D. Notice that with this formulation we can capture the cases of retrieval systems which return documents accompanied by degrees of relevance, since there may be two or more documents with the same degree of relevance.

It can be easily proved that the function *Dist* is still a metric function. Two examples are in order:

- Example (F)

$$
\begin{aligned}
(D)_1 &= < \{d_1, d_2\}, d_3 > \\
(D)_2 &= < d_3, \{d_1, d_2\} > \\
V(d_1) &= 1 + 2 = 3 \\
V(d_2) &= 1 + 2 = 3 \\
V(d_3) &= 2 + 1 = 3 \\
(D)_0 &= < \{d_1, d_2, d_3\} > \\
dist((D)_0,(D)_1) &= 0 + 0 + 1 = 1 \\
dist((D)_0,(D)_2) &= 1 + 1 + 0 = 2
\end{aligned}
$$

Here each document takes the same votes, however notice that $dist((D)_0, (D)_1) \neq$

$dist((D)_0,(D)_2)$! Also note that $dist(D_1,D_2)=3$.

- Example (G)

$$\begin{aligned}(D)_1 &= <\{d_1,d_2\},d_3> \\ (D)_2 &= <d_2,\{d_1,d_3\}> \\ V(d_1) &= 1+2=3 \\ V(d_2) &= 1+1=2 \\ V(d_3) &= 2+2=4 \\ (D)_0 &= <d_2,d_1,d_3> \\ dist((D)_0,(D)_1) &= 0+1+1=2 \\ dist((D)_0,(D)_2) &= 0+0+1=1\end{aligned}$$

Note that $dist(D_1,D_2)=1+0+0=1$.

6 Deriving the Democratic Level of the Fused Ordering

At first, we exploit the function *dist* in order to define the *democratic distance* of the fused ordering $(D)_0$.

Def 6.1 The *democratic distance* of the ordering $(D)_0$, wrt the orderings $(D)_i$ for $i=1..k$, denoted by $Dem((D)_0,\{(D)_1,...,D_k\})$ is given by:

$$Dem((D)_0,\{(D)_1,...,D_k\})=\frac{\sum_{i=1}^{k} dist((D)_0,(D)_i)}{k} \quad (2)$$

◇

Thus it is the average distance between the final ordering $(D)_0$ and each one of the underlying orderings.

The democratic distances of the fused orderings of our examples, are shown in the following table:

(A)	$Dem((D)_0,\{(D)_1,(D)_2\})=2/2$
(B)	$Dem((D)_0,\{(D)_1,(D)_2,(D)_3\})=0/3$
(C)	$Dem((D)_0,\{(D)_1,(D)_2,(D)_3\})=2/3$
(D)	$Dem((D)_0,\{(D)_1,(D)_2,(D)_3\})=4/3$
(E)	$Dem((D)_0,\{(D)_1,...,(D)_6\})=18/6$
(F)	$Dem((D)_0,\{(D)_1,(D)_2\})=3/2$
(G)	$Dem((D)_0,\{(D)_1,(D)_2\})=3/2$

Remark: Notice that although the democratic distance of the final orderings of the Examples (F) and (G) are equal (=3/2), the final ordering in (F) is a set, while the final ordering in (G) is an ordered set.

However recall that our objective is to derive a factor standing for the *level of confidence* (or the democratic level) of the final ordering. This factor can be given to the user in order to distinguish the final orderings which exhibit great degree of homophony, from those with low degree of homophony (or great degree of antiphony). Clearly if the democratic distance is high then the confidence factor should be low. Thus we can derive the confidence factor by appropriately transforming the democratic distance. Let us denote this factor by $CF((D)_0,\{(D)_1,...,D_k\})$. However for notational simplicity we shall use *Dem* for denoting the democratic distance of the final ordering and CF for denoting the confidence factor of that ordering. Two transformations for deriving CF from *Dem* are presented below:

One idea is to employ a linear transformation of the form

$$CF=V-Dem \quad (3)$$

where V is a fixed positive value. Note the V should be big enough if we want CF to be always positive. Thus we can set V equal to the maximum possible democratic distance, that is, when we have antiphony. Note that in case of absolute homophony we get $CF=V$, while in case of absolute antiphony we get $CF=0$. If we want CF to range the interval [0,1], which would be more clear, we can derive CF by the following formula:

$$CF=\frac{V-Dem}{V} \quad (4)$$

Here, in case of homophony we get $CF=1$, while in case of antiphony get $CF=0$.

Another idea, is to employ an inversion transformation of the form

$$CF=V^{-Dem} \quad (5)$$

for some fixed value $V>1$ such as $V=2$ or $V=e$. This transformation provides a measure with a sharp peak at $Dem=0$ gradually sloping away towards 0 as *Dem* becomes larger. For instance consider the Example (E) where we have absolute antiphony with $Dem=3$, and assume that $V=2$. In this case we have $CF=2^{-3}$. Also note that formula (5) releases us from having to compute the maximum possible democratic distance, which depends on the number of documents and the number of systems.

7 Related Work - Concluding Remarks

In general, metasearchers (i.e. MetaCrawler [10], SavvySearch [6], Profusion [3]) merge results from multiple search systems into a single ranked list using some *results fusing* (or merging) strategy. In addition, some form of *query translation* technology is necessary, to interact with different search systems, and some *server selection* method may be available for locating the systems covering documents relevant to the user's query. In this paper we assume that each system accepts the same set of queries, and that the mediator forwards each query to each underlying system, therefore we omit the problems of query translation and server selection, and focus only on the problem of results fusing.

Fusing strategies can be divided into two categories [12]: *integrated methods* and *isolated methods*. Integrated methods require the servers to provide special information for use in fusing, while isolated fusing methods can be applied without any specialized information from servers. Clearly, our technique is an isolated method, since it does not rely on any special information from the underlying systems. This makes our technique generic, since the underlying systems can even employ different methods for indexing and matching.

We are interested in isolated methods because the existing integrated methods except from having narrow applicability, they present some important drawbacks. Commonly, a mediator in order to perform server selection and result merging it takes into account the retrieval effectiveness measures (like Precision/Recall) of the underlying systems ([4]). However, in our opinion, these measures are somehow ill-defined. They are based on the assumption that in a given collection D of documents, and for a given query q, there is a subset R ($R \subseteq D$) of relevant documents, while the rest of the documents ($D \setminus R$) are non-relevant. However, the Information Retrieval problem is based on the assumption that for a given information need, some documents are just *more relevant* than others. Thus the set R cannot be specified exactly. Moreover, there is the problem of judgment: who judges whether a document is relevant to a query, or more relevant than another document ? Certainly there is no all-knowing or widely accepted human or system. This means that in a heterogeneous environment like the web, taking into account these measures, or even comparing the measures of different systems, is certainly an ill-founded approach, and may result to low retrieval effectiveness. For instance, the approach for server selection proposed in [4] presupposes that the mediator knows the number of relevant documents in each underlying system ! In some other approaches (i.e. [12], [13]) they exploit the results of past (or training) queries for estimating the number of relevant documents of each underlying system. However even this approach goes against the autonomy and continuous evolution of the systems of the web.

However, according to our view, the availability of more than one systems offers us a new opportunity: it allows us to derive an aggregated measure of relevance. Thus we can view the problem of results fusing as a group decision problem. Such an approach is better founded and the technique presented in this paper is based on this hypothesis.

Let us now focus on the problem of results fusing. Some approaches assume that the degrees of relevance returned by each system are comparable, and they use them for ordering the results (i.e. [12], [13]), while some others (i.e. [4]) just interleave the returned orderings. In [2] two isolated techniques for merging the search results are introduced. These techniques require downloading the document contents and they also employ a set of relevance collection statistics The drawbacks of these techniques are that the downloading of a document's contents incurs time and bandwidth costs, while the use of relevance collections statistics presupposes that the underlying systems are a-priori known.

Summarizing our discussion we can say that in this paper we presented a results fusing (merging) technique appropriate for environments where there are many heterogeneous and evolving search systems. According to this technique, the fused ordering is derived by aggregating the orderings of the underlying systems using a mathematically sound method which is based solely on the actual results returned by each system. The technique is also enriched by a method for deriving the democratic factor of the fused ordering. Given this factor, the user can draw conclusions about the degree of homophony or antiphony of the results that were returned by the underlying systems as a response to his query. A high factor may drive the user to read only the very first documents of the fused ordering (since they probably are the more relevant to his query), while a low factor may drive him to read more documents.

8 Future Research

An interesting problem for further research is the fusion problem when we allow democratic mediators of other democratic mediators, and so on. If an underlying system is a democratic mediator (as defined in this paper) then this system does not return only an ordered set of documents, but it also returns a confidence factor. This raises two important questions: First, how the confidence factors returned by the subsystems should affect the derivation of the fused ordering, and second, how we should compute the confidence factor of the fused ordering ?

It is also interesting to study the problem of result merging in cases where the mediator knows the "subsystems" of each underlying system. This means that the mediator can find out whether a system (primitive or secondary) has been used more than once. For instance observe that in the mediator $[[S_1, S_2], [S_2, S_3], S_3]$ which is shown graphically in Figure 1, the systems S_2 and S_3 are used more than once.

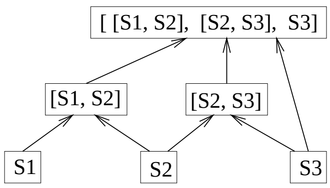

Figure 1: Mediators of mediators of ...

Acknowledgements. Many thanks to Tonia Dellaporta for the fruitful discussions on this work, and to Nicolas Spyratos for his editorial comments.

References

[1] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *"Modern Information Retrieval"*. ACM Press, Addison-Wesley, 1999.

[2] N. Craswell, D. Hawking, and P. Thistlewaite. "Merging Results from Isolated Search Engines". In *Proceedings of the Tenth Australasian Database Conference*, 1999.

[3] Yizhong Fan and Susan Gauch. "Adaptive Agents for Information Gathering from Multiple, Distributed Information Sources". In *1999 AAAI Symposium on Intelligent Agents in Cyberspace*, Stanford University, March 1999.

[4] Norbert Fuhr. "A Decision-Theoretic Approach to Database Selection in Networkded IR". *ACM Transactions on Information Systems*, 17(3), July 1999.

[5] J. R. Giles. *Introduction to the Analysis of Metric Spaces*. Cambridge University Press, 1987.

[6] A. Howe and D. Dreilinger. "SavvySearch: A MetaSearch Engine that Learns Which Search Engines to Query". *AI Magazine*, 18(2), 1997.

[7] Berthier A. Ribeiro-Neto and Richard Muntz. "A Belief Network Model for IR". In *SIGIR'96*, Zurich, Switzerland, 1996.

[8] S. E. Robertson and K. Sparck Jones. "Relevance Weighting of Search Terms". *Journal of the American Society for Information Sciences*, 27(3), 1976.

[9] G. Salton and M. E. Lesk. "Computer Evaluation of Indexing and Text Processing". *Journal of the ACM*, 15(1), January 1968.

[10] E. Selberg and O. Etzioni. "Multi-Service Search and Comparison Using the MetaCrawler". In *Proceedings of the 1995 World Wide Web Conference*, December 1995.

[11] H. R. Turtle and B. W. Croft. "Evaluation of an Inference Netword-Based Retrieval Model". *ACM Transactions on Information Systems*, 9(3):187–222, 1991.

[12] E. Vorhees, N. Gupta, and B. Johnson-Laird. "The Collection Fusion Problem". In *Proceedings of the Third Text Retrieval Conference (TREC-3)*, Gaithersburg, MD, 1995.

[13] Ellen Vorhees. "Multiple Search Engines in Database Merging", 1997. dl 97.