MERGING RESULTS OF DISTRIBUTED IMAGE LIBRARIES

S. Berretti, A. Del Bimbo, P. Pala
Dipartimento Sistemi e Informatica
Università di Firenze

ABSTRACT
Exploitation of information repositories available on the Internet requires users to separately query each repository and manually gather retrieved results. Such a solution could be simplified by using a centralized server that acts as a gateway between the user and repositories: The centralized server forwards the user query to federated repositories and fuses retrieved documents for presentation to the user. To perform these tasks efficiently, the centralized server should perform two main functions: resource selection and data fusion. The former is required to forward the user query only to the repositories that are candidate to contain relevant documents. The latter is used to gather all retrieved documents and conveniently arrange them for presentation to the user. In the case of image repositories, data fusion is particularly challenging owing to the difficulty to normalize document scores returned by different repositories.

In this paper a novel solution is presented for fusion of results returned by different image repositories. Experimental results are presented that show the potential of the proposed approach.

1. INTRODUCTION
Nowadays, many different repositories are accessible through the Internet. A generic user looking for a particular information should contact each repository and verify the presence of the desired information. This typically takes place by contacting the repository server through a client web browsing application. Then, the user is presented a web page that guides her/him in the process of retrieval of information (text documents, images, videos, and so on).

This framework has two major drawbacks. First of all, in order to find the desired information, the user should know in advance at least one repository that store it. Internet search engines such as Altavista, Google and Lycos, to say a few, can support the user in this task. However, if the user doesn’t know the existence of one repository s/he will never contact it (even if it is the only resource storing the desired information). A second drawback is related to the fact that, in order to find the information of interest, the user typically contacts several repositories. Each one is queried and even for a limited number of repositories, the user is soon overwhelmed by a huge and unmanageable amount of (probably irrelevant) retrieved documents.

A much more efficient and convenient solution (especially for the user) is to federate all repositories within a network featuring one central server. A generic user looking for some information sends the query to the server. This one is in charge to forward the query to all (or a subset of all) the networked resources. All retrieved documents (that is the set of all documents returned by each resource) are gathered by the central server and conveniently arranged for presentation to the user.

The implementation of this solution develops upon the availability of three main functions: resource buffering, resource selection and data fusion.

Resource buffering - In general, each resource has its own ways of describing, accessing and presenting stored information. These correspond to different choices in terms of descriptors of content, query paradigms and visualization metaphors. In order to allow the user to seamlessly query several different resources and receive their results, a module is required that acts as a buffer translating information back and forth between the resource and the central server.

Resource selection - In general, it is not always convenient to send the user query to all federated resources. Several considerations can be brought forward to support this decision: sending the query to all resources contribute to network congestion; some resources provide documents for free whereas others request some payment; some resources may have much more relevant documents than others. More precisely, resource selection is the process that, given a user query, computes the number of documents to retrieve from each resource (the query is not sent to resources that should provide no document). An efficient resource selection relies on several information, including cost factors as well as descriptors of content of each resource (resource descriptions).

Data fusion - Since each resource uses its own ways to compute the similarity between its documents and the user query, matching scores of documents returned by different resources are not normalized with each other. This prevents the possibility to infer the relative relevance of documents returned by different resources from the analysis of their matching scores. However, comparing the relative relevance of all retrieved documents is necessary in order to present them in a suitable form to the user. Data fusion is the process by which matching scores of documents returned by different resources are normalized with each other so as to enable comparison of their relative relevance.

In this paper, a novel approach is presented for fusion of results returned by distributed image repositories. The proposed approach relies on learning normalization coeffi-

This work is supported by EU Commission under grant IST-2000-26061. Detailed information about this project is available at http://www.mind-project.org
 cient coefficients. These are used to normalize, with respect to a reference scale, document scores associated by different repositories. This is accomplished through two separate steps. In a first step, each library is processed separately through a set of sample queries. For each sample query, the value of normalization coefficients is computed. In a second step, known values of normalization coefficients are used to estimate their values for new queries.

This paper is organized as follows. In Sect.2, previous work related to data fusion for distributed libraries is reviewed. In Sect.3, the proposed model for data fusion is described. Finally, experimental results are reported in Sect.4.

2. PREVIOUS WORK

Several merging strategies have been proposed to deal with merging results returned by distributed libraries. Most of these approaches apply to text libraries and, for some of them, the extension to deal with multimedia libraries is not straightforward.

One of the most known approaches is called Round-Robin [1]. In this approach, it is assumed that each collection contains approximately the same number of relevant items that are equally distributed on the top of the result lists provided by each collection. Therefore, results merging can be accomplished by picking up items from the top of the result lists in a round-robin fashion (the first item from the first list, the first from the second list, ..., the second from the first list and so on).

Unfortunately, the assumption of uniform distribution of relevant retrieved items is rarely observed in practice, especially if libraries are generic collections available on the Web. As a result, the effectiveness of the Round-Robin approach is very limited.

A different solution develops on the hypothesis that the same search model is used to retrieve items from different collections. In this case, items matching scores are comparable across different collections and they can be used to drive the fusion strategy. This approach is known as Raw Score Merging [2] but is rarely used due to the large number of different search models that are used to index documents even in text libraries.

To overcome limitations of approaches based on raw score matching, more effective solutions have been proposed developing on the idea of score normalization. These solutions assume that each library returns a list of retrieved items with matching scores. Then, some technique is used to normalize matching scores provided by different libraries. Score normalization can also be accomplished by looking for duplicate documents in different lists. The presence of one document in two different lists is exploited to normalize scores in the two lists. Unfortunately, this approach cannot be used if one or more lists are composed only of unique documents, that is documents that are not included in other lists.

A more sophisticated approach based on cross similarity of documents is presented in [4]. For each list, each document is used as a new query. Retrieval results for these new queries as well as matching scores are used to evaluate cross similarities between documents retrieved by different libraries. Then, cross similarities are used to normalize matching scores. The main limitations of this approach are related to its computational complexity (each document in each retrieved list should be used as a new query for all the libraries) and to the fact that it requires the extraction of content descriptors from each retrieved document. This latter requirement that can be easily accomplished for text document, can be quite challenging for image documents.

In order to lessen these requirements, in [5] a new approach is presented that is based on the use of a local archive of documents to accomplish score normalization. Periodically, libraries are sampled in order to extract a set of representative documents. Representative documents extracted from all the libraries are gathered into a central archive. When a new query is issued to the libraries, it is also issued to the central search engine that evaluates the similarity of the query with all documents in the central archive. Then, under the assumption that each retrieved list contains at least two documents that are also included in the central archive, linear regression is exploited to compute normalization coefficient (for each retrieved list) and normalize scores. This approach has been successfully tested for text libraries in combination with query based sampling and CORI [6]. Unfortunately, when this approach is not combined with query based sampling and CORI, the assumption that each retrieved list contains two elements of the central archive is rarely verified. In these cases, linear regression cannot be applied and matching scores are left un-normalized. Furthermore, normalization coefficients should be computed on the fly for each retrieved list. This is associated with additional computational load at retrieval time.

3. DATA FUSION MODEL

The proposed solution develops on the method presented in [5]. Improvements are introduced to address two major limitations of the original method: applicability of the solution to the general case (regardless of the particular solution adopted for resource description); reduction of computational cost at retrieval time. These goals are achieved by decomposing the data fusion process in two separate steps: normalization coefficient learning and normalization coefficient approximation.

Normalization coefficient learning (NCL) is carried out once, separately for each resource: it requires an exchange of information between the resource and the data fusion server. This exchange of information takes place in the form of a set of queries that the data fusion server sends to the resource. For each query, the resource returns a list of retrieved documents with associated matching scores. These are analyzed by the data fusion server that estimates values of the normalization coefficients which transform un-normalized resource matching scores into normalized matching scores.

Differently, normalization coefficient approximation (NCA) is carried out only at retrieval time. It allows, given a query and results provided by a resource for that specific query, to normalize matching scores associated with retrieved results. This is achieved by using information extracted from that resource during NCL.

In doing so, data fusion is achieved through a model learning approach: during NCL, parameters of the model are learned; during NCA, the learned model is used to accomplish data fusion of new results. In the following sections we
describe in detail the model used for data fusion.

### 3.1. Score modeling

Let $L^{(i)}(q) = \{(d_k, s_k)\}_{k=1}^{n_i}$ be the set of pairs document/score retrieved by the $i$-th digital library as a result to query $q$. We assume that the relationship between un-normalized scores $s_k$ and their normalized counterpart $\sigma_k$ can be approximated to a linear relationship. In particular we assume $\sigma_k = a * s_k + b + c_k$, being $c_k$ an approximation error and $a$ and $b$ two parameters that depend both on the digital library (i.e. they may not be the same for two different digital libraries, even if the query is the same) and on the query (i.e. for one digital library they may not be the same for two different queries).

Values of parameters $a$ and $b$ are unknown at the beginning of the learning process. However, during learning, values of these parameters are computed, separately for each library. For each sample query, the value of parameters $a$ and $b$ for different queries is equivalent to sampling the value distribution of these parameters on a grid of points in the query space. Knowledge of parameter values on the grid points is exploited to approximate the value of parameters for new queries. This is accomplished by considering the position of the new query with respect to grid points.

Approximation of parameters $a$ and $b$ for a new query is carried out as follows. If the new query matches exactly one of the sample queries (say $q_k$) used during the learning process, then the value of $a$ and $b$ is set exactly to the value of $a$ and $b$ that was computed for the sample query $q_k$. Otherwise, the three grid points $q_j, q_k, q_l$ that are closest to the new query are considered. The value of $a$ and $b$ for the new query is approximated by considering values of $a$ and $b$ as they were computed for queries $q_j, q_k, q_l$. In particular, if these values were $a_j$, $b_j$, $a_k$, $b_k$, $a_l$ and $b_l$, then values of $a$ and $b$ are estimated as:

$$
a = \frac{d_j}{D} a_j + \frac{d_k}{D} a_k + \frac{d_l}{D} a_l
$$

$$
b = \frac{d_j}{D} b_j + \frac{d_k}{D} b_k + \frac{d_l}{D} b_l
$$

being $D = d_j + d_k + d_l$ and $d_j$, $d_k$ and $d_l$ the Euclidean distances between the new query and grid points $q_j$, $q_k$, $q_l$, respectively.

Using Eqs.1 is equivalent to estimate values of parameters $a$ and $b$ as if the new query lain to the hyperplane passing by $q_j$, $q_k$ and $q_l$. For this assumption to be valid, the new query should be close to the hyperplane. The closer it is, the more precise the approximation.

In the proposed approach, linear regression is the mathematical tool used for the estimation of normalization coefficients, given a sample query and the corresponding retrieved documents. In our case, data points are pairs $(\sigma_i, s_i)$ being $s_i$ the original matching score of the $i$-th document and $\sigma_i$ its normalized score. The application of this method to a collection of pairs $(\sigma_i, s_i)$ results in the identification of the two parameters $a$ and $b$ such that $\sigma_i = a * s_i + b + e_i$, being $e_i$ an error called the residue.

### 3.2. Selection of Sample Queries

Through the NCL phase, the distribution of the value of parameters $a$ and $b$ is sampled for a set of queries. Selection of these sample queries, conditions the quality of the approximation of $a$ and $b$ for a generic query in the query space (during NCA).

Basically, two main approaches can be distinguished for sample query selection. The first approach completely disregards any information available about the distribution of documents in the library. In this case, the only viable solution is to use sample queries uniformly distributed in the query space. A different approach relies on the availability of information about the distribution of documents in the library. This should not be considered a severe limitation since this kind of information is certainly available to accomplish the resource selection task. In particular, this information is available in the form of a resource descriptor capturing the content of the entire library. Typically, resource descriptors are obtained by clustering library documents and retaining information about cluster centers, cluster population and cluster radius. Clustering is performed at several resolution levels so as to obtain multiple cluster sets, each one representing the content of the library at a specific granularity.

Resource descriptors can be used to guide the selection of sample queries by using the cluster center themselves as sample queries. In this way, the distribution of sample queries conforms to the distribution of documents in the library.

Once the resource description process is completed, each cluster center $c_k$ is used as a query to feed the library search engine. This returns a set of retrieved images and associated matching scores $s_k$. Then, $c_k$ is used as a query to feed the reference search engine (i.e. the search engine used by the data fuser) that associates with each retrieved image a normalized (reference) matching score $\sigma_i$. Linear regression is used to compute regression coefficients $(m_k, q_k)$ for the set of pairs $(\sigma_i, s_i)$. In doing so, each cluster center $c_k$ is associated with a pair $(a_k, b_k)$ approximating values of regression coefficients for query points close to $c_k$. The set of points $(c_k, a_k, b_k)$ can be used to approximate the value of regression coefficients $(a, b)$ for a generic query $q$ by interpolating values of $(m_k, q_k)$ according to Eqs.1.

### 4. EXPERIMENTAL RESULTS

The proposed solution to data fusion for distributed digital libraries has been implemented and tested on a benchmark of two different libraries each one including about 1000 images. Experimental results are presented to report both on
the quality of normalization coefficients approximation and on the overall quality of the data fusion process.

Each library was processed separately. Libraries were first subject to the extraction of resource descriptors according to the procedure presented in [7]. Cluster centers extracted through the resource description process were used as sample queries. For each sample query, values of normalization coefficients were computed, as explained in Sect. 3.1. Values of normalization coefficients for sample queries were used to approximate the value of normalization coefficients for new queries. In particular, the quality of the approximation was measured with reference to a random set of test queries (not used during NCL). For each test query, values of normalization coefficients were approximated according to Eqs. 1. Actual values of normalization coefficients were also computed by running the query and following the same procedure used for NCL (Sect. 3.1). Comparison of actual and approximated values of normalization coefficients gives a measure of the approximation quality. Let $a$ and $b$ be the estimated values of normalization parameters and $\hat{a}$ and $\hat{b}$ their actual values. Quality of the approximation is measured by considering the expected values of the normalized error:

$$c_a = E\left[\frac{a - \hat{a}}{1 + |a|}\right], \quad c_b = E\left[\frac{b - \hat{b}}{1 + |b|}\right]$$

Plots in Fig. 2(a) report values of the expected normalized error for two digital libraries and different granularity values of the sampling query grid. Granularity is expressed as percentage of databases population.

To represent the overall quality of the data fusion process, we considered the presence of duplicated images in the retrieved lists. For a generic query, each library returns a list of retrieved documents. The presence of images that are included in both libraries can be used to assess the quality of the data fusion process. In particular, we measure the quality of data fusion as the capability of assigning the same normalized score to duplicate images. Let $L(1)(q) = \{(d_k^{(1)}, s_k^{(1)})\}_{k=1}^n$ and $L(2)(q) = \{(d_k^{(2)}, s_k^{(2)})\}_{k=1}^m$ be the set of pairs document/score retrieved by the two test digital libraries as a result to query $q$. Let the presence of duplicate elements be represented through a function $f: \{1, \ldots, n\} \times \{1, \ldots, m\} \rightarrow \{0,1\}$, that associates with the index of a generic document in the first list either the index of the same document in the second list (if it exists) or the null value.

We assume that the optimal data fusion should associate with duplicate images the same normalized score, that is: $\sigma_i^{(1)} = \sigma_i^{(2)} \forall f(i) \neq 0$. Thus, a measure of the overall quality of the data fusion process is expressed through the following functional:

$$F = \frac{1}{\#\{i = 1, \ldots, n \mid f(i) \neq 0\}} \sum_{i=1, \ldots, n \mid f(i) \neq 0} 2 \frac{2 \sigma_i^{(1)} - \sigma_i^{(2)}}{\sigma_i^{(1)} + \sigma_i^{(2)}}$$

that is, the average sum of the differences of the normalized scores with respect to their average values. The lower the value of $F$ the higher the quality of the data fusion. The value of $F$ is 0 for the optimal data fusion.

Plots in Fig. 2(b) show the value of $F$ averaged on 100 random queries. Values are reported for 5 distinct granularity levels.

5. CONCLUSION AND FUTURE WORK

In this paper, an approach is presented for merging results returned by distributed image digital libraries. The proposed approach is based on learning and approximating normalization coefficients for each library. Future work will investigate the use of more accurate estimates for approximation of normalization coefficients. Further experimentation will also evaluate to what extent the approximation of parameters depends on the distribution of sample queries.

6. REFERENCES


