Database Merging Strategies for Searching Public and Private Collections

Ellen M. Voorhees
National Institute of Standards and Technology
Gaithersburg, MD
ellen.voorhees@nist.gov

A database merging strategy is a method for combining the results produced by searching multiple independent document collections into a single cohesive response. We have devised two such strategies, the Modeling Relevant Document Distributions (MRDD) and Query Clustering (QC) methods, that use the results of past queries to produce a response for the current query [4, 7, 8]. By exploiting historical data, the methods can produce effective merged results with no knowledge of the word frequencies or particular retrieval methodologies of the individual collections. This independence from collection-specific information coupled with the relatively small amounts of historical data the methods must store make the methods appropriate for users who simultaneously search a variety of public and private document collections.

To provide the context for the discussion, the first section of the paper describes the two merging strategies and summarizes the experimental evidence regarding their effectiveness. The second section then contrasts these methods with data-publishing approaches to illustrate how the MRDD and QC methods allow users to customize the merging process to their own searching patterns.

1 The MRDD and QC Database Merging Strategies

1.1 Description

The Modeling Relevant Document Distributions and Query Clustering database merging methods were developed for environments in which search engines work in isolation. The methods assume only that the document collections available to be searched accept a query in natural language and return a ranked list of documents in response. Both methods require a set of training queries — queries for which the user has judged documents as relevant or nonrelevant across (a subset of) the collections to be searched — and produce a single ranked list of documents in response to new queries.

A complete solution to the database merging problem requires a solution to two subproblems: the resource selection problem (i.e., which collections to search) and the data fusion problem (i.e., how to combine the results of the collections that were searched). The MRDD and QC methods differ only in how they select which databases to search.

In the training phase, the MRDD method represents the relevant document distribution of each training query in each collection by the ranks obtained by the relevant retrieved documents. It also builds a vector space from the text of the training queries. To process a new query q, it uses the vector space to compute the k nearest neighbors of q. For each collection, it builds an average relevant document distribution for q by taking the mean of the distributions of the k nearest neighbors. It passes these averaged distributions to an optimization procedure that computes the number of documents to retrieve from each collection such that N documents total are retrieved and the number of relevant retrieved is maximized.

Instead of explicitly storing the relevant document distribution for each training query, the QC method summarizes query retrieval performance for a collection in a set of clusters. In the training phase, the QC method clusters the training queries in each collection using the number of documents retrieved in common as a similarity metric. A centroid vector is built for each cluster, and each cluster is assigned a weight of the average number of relevant documents the queries in the cluster retrieved. To process new query q, the centroid that is closest to q is found in each collection and the corresponding cluster weight is returned. The number of documents to retrieve from each collection is computed as $\frac{w_i}{\sum_{i=1}^{C} w_i} * N$ where N documents total are to be retrieved and w_i is the weight returned by collection i.

For both methods, once the desired number of documents are retrieved from each collection, a final ranked list must be produced. Since the similarity scores computed in different collections are incomparable, they cannot be used to form a single total order on the documents that have been returned. Instead, the ranking is created probabilistically, one rank at a time. To select the document to be placed at the next rank r, a collection is randomly selected and the next document in that collection's result is placed in r. The probability of selecting a collection is proportional to the number of documents it has remaining to be placed. The advantages of this ranking scheme are that it obeys the orderings produced by the individual collections and it gives preference to documents from collections with many retrieved documents.

1.2 Effectiveness

The effectiveness of the MRDD and QC methods has been tested in a series of retrieval experiments using the TREC data set [4, 5, 6, 7, 8]. In all of the experiments the goal has been to produce a merged result that is at least as effective as the result obtained if the entire set of documents is treated as a single collection. For retrieved set sizes of interest to most users (50 documents retrieved per query or fewer), the merged search results are generally within 15% of the single collection results in these tests. A 15% degradation amounts to approximately one fewer relevant document retrieved per query.

Various retrieval experiments tested conditions the merging strategies would face in an operational setting. Tests using two quite different search engines, TOPIC and SMART, confirm that the behavior of the MRDD and QC techniques is the same regardless of which engine, or combination of engines, is used [8]. Tests in which different collections had different training sets demonstrated that the methods do not require all training queries to be run against all collections for the merging to be successful [5]. These same tests also showed that the QC method is more robust in the face of lower quality training data than the MRDD method.

2 Facilitating Searches of Private and Public Collections

A common approach to the database merging problem is to assume that collections will publish data such as term frequency statistics to facilitate resource selection. The collections may each be subject to some central authority that mandates the data sharing, such as in a business intranet, or may voluntarily agree to publish the data using a standard format such as the format suggested by the Stanford Protocol Proposal for Internet Retrieval and Search (STARTS) [2]. Some solutions to the merging problem are known for these cases. For example, the *gGlOSS* system uses the data made available in the STARTS protocol to select collections [3], and Callan and his colleagues show how an inference net can use term document frequencies to rank document collections [1].

However, the data-publishing approach is limited to those environments in which *all* collections publish the required data. In environments in which users must search large, public collections as well as in-house, proprietary, or personal collections to satisfy their information needs, it is highly unlikely that all of the collections will publish the same collection data. While collection providers have incentive to publish some statistics (to generate more users), there are strong disincentives as well. Making term statistics available in an external form adds to the storage overhead of the collection and adds another facet to collection maintenance. Word frequency statistics can expose too much information regarding a proprietary collection. Furthermore, if all the collections *did* publish the appropriate statistics, the user must then gather and maintain a statistics database since it would be prohibitively slow to gather the appropriate statistics from each collection at query run time. (Alternatively, the user could rely on a central site to gather the appropriate statistics and do the merging. However, the central site would presumably not have access to the collection statistics of the in-house, proprietary, and personal collections.)

The MRDD and QC methods allow a user to customize the merging process to cover precisely the set of collections desired, called the user's universe of collections. Any collection the user has access to can belong to his or her universe since all that is required of a collection is that it respond to queries. Maintenance of the metadata needed to decide which collections in the universe should be searched for a particular query is handled by the training process that is part of the merging strategy. An important feature of this metadata is that each collection's data is independent from the others. This feature enables users to delete collections from their universe without re-training the merging methods. Collections can be added to the universe as well, though some accommodation for training on that collection must be made.

In an environment where the collections are dispersed in a large network, the time to process a query will be dominated by the amount of communication required at run time between the search sites and the merging site. Both the MRDD and QC methods use the locally stored metadata to decide how many documents to retrieve from each collection. Therefore, only those collections that actually contribute documents to the final result are contacted at run time. Note that if the collections cover different subject areas, only a few collections will be contacted for each query since the methods will decide to retrieve zero documents from the majority of collections.

3 Conclusion

The Modeling Relevant Document Distributions (MRDD) and Query Clustering (QC) database merging methods were designed for environments in which search engines work in isolation. A major tenet of the isolation strategy is that the site responsible for the merging should incur all processing and storage costs of the merge. In addition to eliminating the need for cooperation from the search sites, this tenet forces the metadata to be stored locally, which is key to the efficiency of the methods. The QC and MRDD methods are thus appropriate for users who routinely search a combination of public and private collections.

Both methods rely on training data to avoid collection dependencies. As with any learning method, this makes the effectiveness of the methods dependent on the quality of the training data. Thus the methods are not appropriate for use in environments where collection content changes quickly and dramatically, or for environments where queries are uncorrelated with one another.

References

[1] James P. Callan, Zhihong Lu, and W. Bruce Croft. Searching distributed collections with inference networks. In Edward A. Fox, Peter Ingwersen, and Raya Fidel, editors, *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 21–28, July 1995.

[2] Luis Gravano, Kevin Chang, Hector Garcia-Molina, Carl Lagoze, and Andreas Paepcke. STARTS: Stanford protocol proposal for internet retrieval and search. Technical report, Digital Library Project, Stanford University, November 1993. http://www-db.stanford.edu/~gravano/starts.ps.

[3] Luis Gravano and Hector Garcia-Molina. Generalizing *GlOSS* to vector-space databases and broker hierarchies. In *Proceedings of the 21st VLDB Conference*, 1995.

[4] Geoffrey Towell, Ellen M. Voorhees, Narendra K. Gupta, and Ben Johnson-Laird. Learning collection fusion strategies for information retrieval. In *Proceedings of the 12th Annual Machine Learning Conference*, July 1995.

[5] Ellen M. Voorhees. Siemens TREC-4 report: Further experiments with database merging. In Donna K. Harman, editor, *The Fourth Text REtrieval Conference (TREC-4)*, pages 121–130, October 1996. NIST Special Publication 500-236.

[6] Ellen M. Voorhees, Narendra K. Gupta, and Ben Johnson-Laird. The collection fusion problem. In Donna K. Harman, editor, *Overview of the Third Text REtrieval Conference (TREC-3) [Proceedings of TREC-3.]*, pages 95–104, April 1995. NIST Special Publication 500-225.

[7] Ellen M. Voorhees, Narendra K. Gupta, and Ben Johnson-Laird. Learning collection fusion strategies. In Edward A. Fox, Peter Ingwersen, and Raya Fidel, editors, *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 172–179, July 1995.

[8] Ellen M. Voorhees and Richard Tong. Multiple search engines in database merging. In *Proceedings of the 2nd ACM International Conference on Digital Libraries*, July 1997. To appear.