Merging Retrieval Results in Hierarchical Peer-to-Peer Networks

Jie Lu
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213, USA
jielu@cs.cmu.edu

Jamie Callan
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213, USA
callan@cs.cmu.edu

Categories and Subject Descriptors
H.3.3 [**Information Search and Retrieval**]: Retrieval models, Search process, Selection process.

General Terms
Algorithms, Performance, Experimentation, Design.

Keywords
Peer-to-peer, Hierarchical, Retrieval, Result Merging.

1. INTRODUCTION
Peer-to-peer (P2P) networks are an appealing approach to federated search over large networks of digital libraries. The nodes in peer-to-peer networks can participate both as servers that provide information and as clients that request information. Hierarchical P2P architectures introduce special nodes called "hubs". Each hub provides regional directory services for portions of the network; hubs collectively cover the whole network. The directory services provided may include routing information requests ("resource selection") and combining the retrieval results from different digital libraries ("leaf nodes") into a single, integrated ranked list ("result merging"). Viewing peer-to-peer networks as a particular type of distributed IR environment, we explore content-based retrieval in hierarchical P2P networks where digital libraries may not cooperate or may have an incentive to cheat, and hubs provide result merging services to combine retrieval results from multiple resources. In this environment, digital libraries do not need to provide the information about their contents to hubs. Instead, hubs use techniques such as query-based sampling to discover the contents of neighboring digital libraries for query routing as well as result merging.

2. RESULT MERGING ALGORITHMS FOR HIERARCHICAL P2P NETWORKS
Figure 1 illustrates content-based retrieval in hierarchical P2P networks. The C (white) node is the client node that issues the information request, the H (black) nodes are hubs, and the D (gray) nodes are leaf nodes (digital libraries). The edges between nodes represent connections and the arrows on the edges indicate the directions to send back retrieval results to the client node.

In hierarchical P2P networks, result merging naturally takes place at hubs because they already provide directory services to route information requests; the information they maintain for resource selection may also be useful for result merging. The results that need to be merged at a hub (e.g. H_1) may include not only the results from the neighboring leaf nodes (e.g. D_1), but also the results sent back by other hubs down the query path (e.g. H_3).

Result merging may also take place at the client node if it issues the request to more than one hub. Because client nodes don't maintain information about the contents of digital libraries as hubs do, they can use only simple, but probably less effective, merging methods, e.g. merging results by document scores returned from these hubs ("raw score merge") or in a round robin fashion.

In this paper, we explore a set of result merging algorithms that can be used by hubs. These algorithms use *centralized sample databases* at hubs but differ in how to use them. A *centralized sample database* at a hub is composed of documents obtained by query-based sampling of neighboring leaf nodes and aggregate statistics provided by neighboring hubs.

2.1 Semi-Supervised Learning
The Semi-Supervised Learning (SSL) result merging algorithm was initially developed for distributed IR environments with a single directory service. We adapted it to hierarchical P2P networks with multiple hubs. A hub that uses SSL learns a query-dependent score normalizing function for each of its neighboring nodes. This function transforms neighbor-specific document scores to hub-specific document scores. The (unsupervised) training data needed by SSL is pairs of neighbor-specific and hub-specific scores for a set of documents [4]. An effective resource for generating this training data is the centralized sample database at the hub. Given a query, the overlapping documents between the retrieval results from the centralized sample database at the hub and those from a neighboring node are identified and their scores serve as training data for learning the score normalizing function.

Typically, SSL uses linear regression to learn score normalizing functions, which does not work as well on our peer-to-peer testbed [2] as on traditional distributed IR testbeds. The tradeoff between accuracy and communication costs for retrieval in P2P networks often leads to too few overlapping documents as training data, more biased resource representations from sampled documents, and nonlinear score transformation, which greatly affects the performance of the algorithm in P2P environments. We developed two new solutions to address the above problems. One solution modifies SSL to learn more sophisticated score normalizing functions. We introduce it in the following section.

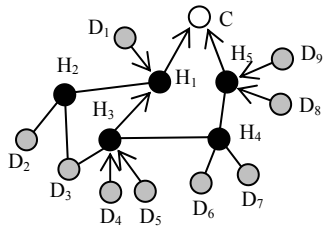

Figure 1 Retrieval in Hierarchical P2P Networks.

SIGIR'04, July 25–29, 2004, Sheffield, South Yorkshire, UK.
ACM 1-58113-881-4/04/0007.

The other solution directly estimates hub-specific document scores but requires some degree of cooperation from leaf nodes. Section 2.3 presents it in more detail.

2.2 Modified Semi-Supervised Learning

SSL was modified as follows:

1. Locally weighted linear regression is applied to learn the score transformation when the number of training points in the neighborhood is at least *N* and the operation is interpolation instead of extrapolation; otherwise, globally non-weighted linear regression is applied.
2. As is described in [4], downloading more documents to serve as additional training data whenever needed could improve the performance of SSL. Thus when the number of overlapping documents is less than *T* for a neighboring node, up to *D* documents in the retrieval results are downloaded from this node and added to the set of overlapping documents.
3. When the score normalizing function learned using globally non-weighted linear regression has negative slope, the documents in the retrieval results are downloaded one at a time as additional training data until the slope is positive.

N, *T*, and *D* are parameters of the modified SSL algorithm. The values used in our experiments were 3, 2, and 3 respectively.

2.3 Score Estimation with Sample Statistics

Our new Score Estimation with Sample Statistics (SESS) algorithm directly calculates hub-specific document scores using the Kullback-Leibler divergence-based retrieval algorithm [3]. Ordinarily retrieval algorithms are applied to the documents themselves, but our goal is to merge document rankings prior to deciding which (if any) documents to download.

SESS is a cooperative algorithm that requires neighbors to provide summary statistics for each of their top-ranked documents, for example, document lengths and how often each query term matched. This can be viewed as an extension of Kirsch's algorithm [1]. It allows very accurate normalized document scores to be determined before downloading any documents.

Although the SESS algorithm requires cooperation, which we would prefer to avoid, it is a limited degree of cooperation that only improves efficiency. A hub could acquire the same information uncooperatively, at greater cost, by downloading the top-ranked documents from neighboring nodes.

SESS doesn't require neighboring nodes to provide query term corpus frequencies, which makes it different from other algorithms that require full cooperation from participants. It uses sample statistics generated from the centralized sample database at the hub to approximate these corpus statistics.

3. EVALUATION

We used the same P2P testbed as in [2] and 1,500 queries randomly selected from the set of 15,000 automatically-generated queries used in [2]. Because it is expensive to obtain relevance judgments for so many automatically-generated queries, 50 top-ranked documents retrieved using a single large collection (the subset of the WT10g used to define leaf node contents in the P2P network) were treated as the relevant documents for each query. The merged retrieval results from the hierarchical P2P networks were evaluated using 11-point recall-precision.

Figure 2 shows the 11-point recall-precision curves for different result merging algorithms. The upper bound was generated by the documents that were returned by retrieval in the P2P network and sorted by their corresponding scores in the retrieval results from the single WT10g-subset collection. The lower bound was generated by directly merging documents from different leaf nodes using their initial document scores ("raw score merge").

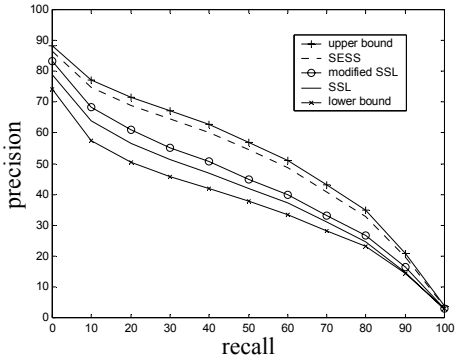

Figure 2 The 11-Point Recall-Precision Curves for Different Result Merging Algorithms.

The SSL algorithm improved the average precision by 11.6% compared with the lower bound. Compared with the original SSL algorithm, modified SSL had an 8.6% relative performance gain in average precision. Modified SSL also worked quite well at low recall levels compared with the upper bound.

SESS worked much better than SSL and modified SSL. In fact, SESS had near optimal performance compared with the upper bound. The relative performance loss in average precision was 4.9%, which few users would notice. The gap between modified SSL and SESS is the value of cooperation. A small amount of cooperation from neighboring nodes greatly improves result merging accuracy, without significant communication costs.

4. CONCLUSIONS

Result merging in hierarchical P2P networks is not a simple adaptation of existing approaches due to the unique characteristics of P2P environments, for example, multiple levels of result merging, skewed collection statistics, and higher communication costs. The Semi-Supervised Learning (SSL) algorithm is modified and a new algorithm Score Estimation with Sample Statistics (SESS) is proposed. Experimental results demonstrate that SESS is very effective in hierarchical P2P networks and modified SSL has satisfactory precision for top-ranked merged documents.

ACKNOWLEDGMENTS

This material is based on work supported by NSF grants IIS-00118767 and IIS-0240334. Any opinions, findings, conclusions or recommendations expressed in this material are the authors', and do not necessarily reflect those of the sponsor.

REFERENCES

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

[2] J. Lu and J. Callan. Content-cased retrieval in hybrid peer-to-peer networks. In *Proc. of the 12nd International Conference on Information Knowledge Management*, 2003.

[3] P. Ogilvie and J. Callan. Experiments using the Lemur toolkit. In *Proc. of the 10th Text Retrieval Conference*, 2001.

[4] L. Si and J. Callan. A semi-supervised learning approach to merging search engine results. *ACM Transactions on Information Systems*. To appear.