Challenges of Distributed Search Across Digital Libraries

Matthias Bender, Sebastian Michel, Gerhard Weikum, Christian Zimmer
{mbender, smichel, weikum, czimmer}@mpi-sb.mpg.de

Max-Planck-Institut für Informatik, 66123 Saarbrücken, Germany

Abstract. We present the MINERVA[2] project that tackles the problem of collaborative search across a large number of digital libraries. The search engine is layered on top of a Chord-style peer-to-peer overlay network that connects an a-priori unlimited number of peers or digital libraries. Each library posts a small amount of metadata to a conceptually global, but physically distributed directory. This directory is is used to efficiently select promising libraries to execute a query based on their local data. The paper discusses current challenges regarding replication, caching and proactive dissemination, query routing based on local user profiles such as bookmarks, and benefit/cost models for query routing.

1 Introduction

The peer-to-peer (P2P) approach allows handling huge amounts of data of digital libraries in a distributed and self-organizing way. These characteristics offer enormous potential benefit for search capabilities powerful in terms of scalability, efficiency, and resilience to failures and dynamics. Additionally, such a search engine can potentially benefit from the intellectual input (e.g., bookmarks, query logs, click streams, etc.) of a large user community. However, recent research on structured P2P architectures is typically limited to exact-match queries on keys. This is insufficient for text queries that consist of a variable number of keywords, and it is absolutely inappropriate for full-fledged Web search where keyword queries should return a ranked result list of the most relevant approximate matches [7].

This paper builds upon the MINERVA system architecture presented in [4] and brings current challenges to ultimatively making distributed search across digital libraries feasible. MINERVA provides ranked search on data and and an efficient query mechanism that adheres to reasonable space and bandwidth limits. Unlike the approach criticized in [10], we do not spread inverted lists across the directory, but use only pointers to promising digital libraries as compact metadata and utilize these pointers to efficiently answer multiple-keyword queries. We leverage the extensive local indexes (which would be impossible to efficiently share across all peers) to incorporate features that are impossible in the approach studied in [10], such as phrase matching or proximity searches. Our

[2] Roman goddess of science, wisdom, and learning

bandwidth requirements are well within the postulation that a query should send no more data than the size of the documents ultimatively retrieved.

2 Related Work

Recent research on structured P2P systems, such as Chord [17], CAN [13], Pastry [15], or P-Grid [1] is typically based on various forms of distributed hash tables (DHTs) and supports mappings from keys to locations in a decentralized manner such that routing scales well with the number of peers in the system. In the following we briefly discuss some prior and ongoing projects towards P2P Web search.

Galanx [19] is a P2P search engine implemented using the Apache HTTP server and BerkeleyDB. The Web site servers are the peers of this architecture; pages are stored only where they originate from. PlanetP [8] is a publish-subscribe service for P2P communities, supporting content ranking search. PlanetP distinguishes local indexes and a global index to describe all peers and their shared information. The global index is replicated using a gossiping algorithm. The system appears to be limited to a few thousand peers.

Odissea [18] assumes a two-layered search engine architecture with a global index structure distributed over the nodes in the system. A single node holds the complete, Web-scale, index for a given text term (i.e., keyword or word stem). Query execution uses a distributed version of Fagin's threshold algorithm [9]. The system appears to cause high network traffic when posting document metadata into the network, and the presented query execution method seems limited to queries with at most two keywords. The paper actually advocates using a limited number of nodes, in the spirit of a server farm. The system outlined in [14] uses a fully distributed inverted text index, in which every participant is responsible for a specific subset of terms and manages the respective index structures. Particular emphasis is put on minimizing the bandwidth used during multi-keyword searches. [11] considers content-based retrieval in hybrid P2P networks. The peer selection for forwarding queries is based on the Kullback-Leibler divergence between peer-specific statistical models of term distributions.

In addition to this recent work on P2P search, prior research on distributed IR and metasearch engines is also potentially relevant; see [6, 20] for overviews. However, this work has assumed a relatively small number of digital libraries or databases and a fairly static configuration.

3 System Design

As a detailed description of the system design has already been given in [4], we only present a brief overview here and refer the interested reader to this prior work.

We view every library as autonomous. A conceptually global but physically distributed directory, which is layered on top of a Chord-style dynamic hash table (DHT) [17], holds only very compact, aggregated information about the peers' local indexes and only to the extent that the individual peers are willing to disclose. Every peer is responsible for a randomized subset of the global directory.

The global directory consists of aggregated information (*Posts*) that contains contact information about the digital library who posted this summary together with statistics to calculate IR-style relevance measures that try to estimate the relevance of a particular digital library to a query. These measures are used to support the *peer selection* process, i.e., determining the most promising libraries for a particular query.

If, at query time, the local query result is considered unsatisfactory by the user, a library retrieves a list of potentially useful libraries. Using collection selection methods from distributed IR and metasearch, a number of promising libraries for the complete query is computed from these PeerLists. In [3] we have studied promising and efficient techniques for this purpose. Subsequently, the query is forwarded to these libraries and executed based on their local indexes. The results from the various libraries are combined at the querying peer into a single result list; this step is referred to as *result merging*.

The goal of finding high-quality search results with respect to precision and recall cannot be easily reconciled with the design goal of unlimited scalability, as the best information retrieval techniques for query execution rely on large amounts of document metadata. Posting only compact, aggregated information about local indexes and using appropriate peer selection methods to limit the number of peers involved in a query keeps the global directory manageable and reduces network traffic. We expect this approach to scale very well as more and more peers jointly maintain the moderately growing global directory.

4 Challenges

Our work is driven by the question of how a collaborative search acorss digital libraries can benefit from the unique nature of the P2P paradigm. We want to address the shortcomings of centralized search engines and further benefit from the intellectual input of a large user community.

To overcome the danger of the infiltration of a centralized index by (commercial) interest groups (as increasingly seen today on popular web search engines) we leverage the indexes from potentially thousands of digital libraries, making it harder to bias the final query results. Additionally, we want to incorporate the fact that every library has its own local index, e.g., by executing all queries first locally at the initiating library and using implicit-feedback techniques for automated query expansion (e.g., using the well-known IR technique of pseudo relevance feedback [5] or other techniques based on query logs [12] and click streams [16]).

We want to incorporate local user bookmarks into our query execution [2]. Bookmarks represent strong recommendations for specific documents. Also, user bookmarks can be considered as compact samples of peer indexes that describe their fields of interest. Queries could be exclusively forwarded to thematically related libraries with similarly interested users, to improve the chances of finding *subjectively* relevant pages.

In order to achieve service levels comparable to today's search engines and in order to actually benefit from the infrastructural advantage of a distributed system, we are going to introduce a certain degree of replication to the directory.

Currently, as dictated by the DHT-style maintenance approach, a peer gracefully leaving the system forwards its share of the global directory to another peer. Analogously, a library entering the system asks for its appropriate share of the directory. While one approach to ensure a valid, complete, and up-to-date directory is to simply rely on the libraries to regularly re-send their Posts to the directory, it is also necessary to replicate parts of the directory to avoid data loss as libraries ungracefully leave the system. Also, replication can serve as a load-balancing measure, relieving the burden from peers that host popular parts of the directory, and increases data availability. Adaptive, self-tuning strategies for choosing appropriate degrees of replication and placing replicas are an open issue in our ongoing work.

To further enhance query efficiency, statistical summaries and also the results of queries from across the network may be cached in a way that allows other peers not only to instantly benefit from the existing query results but also to benefit from click streams that were recorded on the occasion of similar queries. Statistical summaries may also be disseminated proactively among thematically related peers. Finding good strategies to this end is a widely open issue.

Peer selection has been a research issue for years. Most of the existing literature estimates the expected result quality of a collection, typically using precomputed statistics, and ranks the collections accordingly. We believe that this is insufficient if the collections overlap, e.g., in the scenario of digital libraries that share an arbitrarily large fraction of documents. We argue for the extension of existing quality measures using estimators of mutual overlap among collections. Preliminary experiments show that such a combination can outperform popular approaches based on quality estimation only, such as CORI [6]. Taking overlap into account during collection selection in this scenario can drastically decrease the number of libraries that have to be contacted in order to reach a satisfactory level of recall, which is a great step towards the feasibility of distributed search across digital libraries.

Ultimatively, we want to introduce a sophisticated $benefit/cost$ ratio when selecting remote libraries for query forwarding. For the benefit estimation, it is intuitive to consider such measures as described in the previous paragraph. Defining a meaningful cost measure, however, is an even more challenging issue. While there are techniques for observing and inferring network bandwidth or other infrastructural information, expected response times (depending on the current system load) are changing over time. One approach is to create a distributed Quality-of-Service directory that, for all peers, holds moving averages of recent response times.

References

1. K. Aberer, M. Punceva, M. Hauswirth, and R. Schmidt. Improving data access in p2p systems. *IEEE Internet Computing*, 6(1):58–67, 2002.
2. M. Bender, S. Michel, G. Weikum, and C. Zimmer. Bookmark-driven query routing in peer-to-peer web search. In J. Callan, N. Fuhr, and W. Nejdl, editors, *Proceedings of the SIGIR Workshop on Peer-to-Peer Information Retrieval*, pages 46–57, 2004.

3. M. Bender, S. Michel, G. Weikum, and C. Zimmer. The minerva project: Database selection in the context of p2p search. *Accepted to BTW05; Available at http://www.mpi-sb.mpg.de/units/ag5/software/minerva/ publications/submitted05.pdf*, 2005.
4. M. Bender, S. Michel, C. Zimmer, and G. Weikum. Towards collaborative search in digital libraries using peer-to-peer technology. In *DELOS Workshop: Digital Library Architectures*, pages 61–72, 2004.
5. C. Buckley, G. Salton, and J. Allan. The effect of adding relevance information in a relevance feedback environment. In *Proceedings of the seventeenth annual international ACM-SIGIR conference on research and development in information retrieval*. Springer-Verlag, 1994.
6. J. Callan. Distributed information retrieval. *Advances in information retrieval, Kluwer Academic Publishers.*, pages 127–150, 2000.
7. S. Chakrabarti. *Mining the Web: Discovering Knowledge from Hypertext Data*. Morgan Kaufmann, San Francisco, 2002.
8. F. M. Cuenca-Acuna, C. Peery, R. P. Martin, and T. D. Nguyen. PlanetP: Using Gossiping to Build Content Addressable Peer-to-Peer Information Sharing Communities. Technical Report DCS-TR-487, Rutgers University, Sept. 2002.
9. R. Fagin. Combining fuzzy information from multiple systems. *J. Comput. Syst. Sci.*, 58(1):83–99, 1999.
10. J. Li, B. Loo, J. Hellerstein, F. Kaashoek, D. Karger, and R. Morris. On the feasibility of peer-to-peer web indexing and search. In *In 2nd International Workshop on Peer-to-Peer Systems (IPTPS)*, 2003.
11. J. Lu and J. Callan. Content-based retrieval in hybrid peer-to-peer networks. In *Proceedings of CIKM03*, pages 199–206. ACM Press, 2003.
12. J. Luxenburger and G. Weikum. Query-log based authority analysis for web information search. In *WISE04*, 2004.
13. S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Schenker. A scalable content-addressable network. In *Proceedings of ACM SIGCOMM 2001*, pages 161–172. ACM Press, 2001.
14. P. Reynolds and A. Vahdat. Efficient peer-to-peer keyword searching. In *Proceedings of International Middleware Conference*, pages 21–40, June 2003.
15. A. Rowstron and P. Druschel. Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems. In *IFIP/ACM International Conference on Distributed Systems Platforms (Middleware)*, pages 329–350, 2001.
16. J. Srivastava, R. Cooley, M. Deshpande, and P.-N. Tan. Web usage mining: Discovery and applications of usage patterns from web data. *SIGKDD Explorations*, 1(2):12–23, 2000.
17. I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan. Chord: A scalable peer-to-peer lookup service for internet applications. In *Proceedings of the ACM SIGCOMM 2001*, pages 149–160. ACM Press, 2001.
18. T. Suel, C. Mathur, J. Wu, J. Zhang, A. Delis, M. Kharrazi, X. Long, and K. Shanmugasunderam. Odissea: A peer-to-peer architecture for scalable web search and information retrieval. Technical report, Polytechnic Univ., 2003.
19. Y. Wang, L. Galanis, and D. J. de Witt. Galanx: An efficient peer-to-peer search engine system. *Available at http://www.cs.wisc.edu/ yuanwang*.
20. C. Yu, W. Meng, K.-L. Liu, W. Wu, and N. Rishe. Efficient and effective metasearch for a large number of text databases. In *Proceedings of CIKM99*, pages 217–224. ACM Press, 1999.