Parameters in P2P-IR

Wolfgang Müller, Andreas Henrich
LS Medieninformatik, Universität Bamberg
wolfgang.mueller@wiai.uni-bamberg.de

Abstract: P2P Information Retrieval (P2P-IR) is a challenging and potentially very useful area of research. Its goal is to enable powerful, efficient, massively distributed IR. A large fraction of current research is focused on efficiency. However, we feel that P2P-IR needs to address the complicated combination of indexing structure and networking issues.

1 Introduction

P2P-IR algorithms need to achieve two things at the same time. *Firstly,* they need to maintain a structural invariant in the network in the presence of peers joining and leaving the network. *Secondly,* they need to use the structural invariant in order to perform efficient query processing.

In P2P-IR it is still visible that the area is interdisciplinary with participants from two, maybe three communities. *Firstly,* the *networking community*, *secondly,* the data base community, *thirdly,* the information retrieval community. We keep the second and the third distinct, as there is a major difference: the data base community is mainly interested in *exact querying,* while the information retrieval community is more focused on the *user*, his information need and the fact that while perfection is hard to achieve in many research areas, in IR it is even hard to define what perfection *is*.

In current research we see three main architectural groups. *Firstly,* Distributed-Hash-Table-based[1] distributed indexing structures [TXM03], *secondly,* unstructured systems that make use of replication of index data in order to achieve efficiency [SBR04, YGM03], and, *thirdly,* summary based semantic routing systems that achieve efficiency by data source selection [MH03, CAN02, TSW04].

Within this paper we turn our attention to factors that determine efficiency and result quality of P2P information retrieval. Systems that use different architectural paradigms will tend to respond differently to the challenges posed by the system environment. Concluding the paper we will derive a small set of interesting research directions that we feel are interesting directions for future research in P2P-IR.

The general idea of this article is that we feel that P2P-IR should turn its attention to all parts of a peer's life cycle. Much of the current research in P2P-IR appears to focus on

[1] For literature about DHTs, see [RD01, SMK+01, RFH+01]

the *operation* of the P2P-IR network. However, as we will argue below, there is a need to also look at how to make *join* and *leave* efficient in order to cope with peers that enter the network only for a short time.

2 Churn

The popularity of Napster, the legally problematic motivation for the current wave of research on P2P networks, was due to a number of factors. The main factor was the product *"music for free"*, however, music files were offered by other services such as mp3.com or ftp servers. Napsters competitive edge was the ease of use and the *freshness* of data. Napster's index data reflected closely the current state of resources in the system. As documents were provided by a large number of data sources, it was critical to stay informed about data sources and their online state [Men03].

P2P networks tried to mimck and extend Napster's advantages without making use of a central indexing server. So, P2P networks try to be administration-free, highly available and in addition to that to provide freshness of index data. In order to do that, they have to cope with *churn*. Churn is the coming and going of peers within the system.

We see three interesting effects that will influence algorithms that try to cope with churn while providing retrieval performance.

Short term churn: session duration Short term churn denotes events that happen on a scale of minutes and hours. Such effects have been studied by several authors [KLVW04, BSV03]. While these publications differ in detail, they approximately support the data from [KLVW04] that are summarised as follows:

- The session time (time between coming of leaving) of 80% is below 2 minutes.
- Half of the remaining 20% stay more than 50 minutes in the network.

So, these data suggest that P2P protocols have to cope with a large number of peers that connect only very shortly. We will call these peers *cameo peers* in the following.

Long term churn: comeback probabilities In addition to the short term effects described in the previous section, there are long-term churn effects. A user who subscribed to a peer will not come back indefinitely, *e.g.* every day, but there is a possibility that he will not come back. [BSV03] saw 20% of the nodes departing every day in the 14 days they did measurements.

A high comeback probability allows us to cache data of other peers that we expect back at some moment. A low comeback probability makes such caching schemes useless, as much data will be cached for little effect.

Time of day effects Several publications (*e.g.* [KLVW04]) have observed time of day effects. The distribution of internet-literate people over the world is extremely uneven. So, depending on the time of day, the number of people online in a world-spanning P2P system varies strongly.

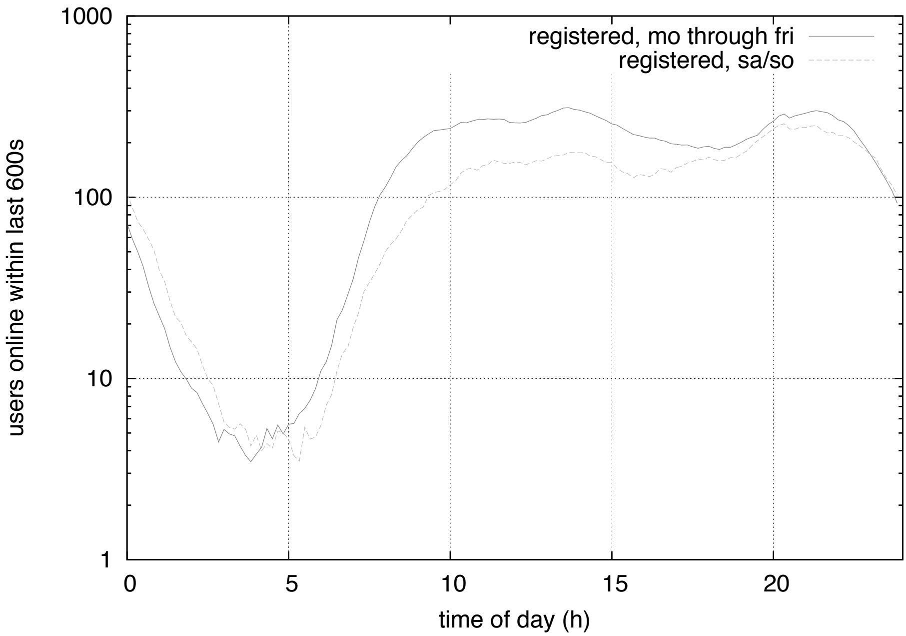

Abbildung 1: Time of day effects in a large bulletin board.

A trivial consequence of time-of-day effects is that it makes the construction of P2P services whose interest is only national or even regional (*e.g.* a specialized collection of Bavarian songs) very difficult, if not impossible. Fig. 1 might serve as an illustration. It represents time-of-day usage statistics of a popular german bulletin board of young parents that has at the time of writing 18'000 subscribed users. While the application in question is not a search application it does give an impression of online times. As participating in the board is perfectly legal, there is no incentive to leave the network quickly.

3 Data

An important factor for the efficiency of P2P information retrieval methods are the data to be indexed. We will consider that we have four *kinds*[2] of data in the system: *Documents:* The data to be indexed and retrieved by the system. *Index data:* Automatically extracted, verbose features of each document. These can be of the same size as the indexed documents. *Meta data:* Data that is assigned by hand to a document with the goal of indexing a document by that data. These data usually are terse compared to index data. *(Peer) summa-*

[2] As opposed to data *types*.

ries: Automatically derived summaries that summarise the data present in a peer. Typically summaries of *one peer* have about the size of the index data for *one document*.

A P2P system's performance is determined by the data kinds, as well as by the location of actual chunks of such data.

Data types and data structures To a large extent, the performance of a P2P system is determined by the data type of the index data, as this influences the indexing structure chosen and the query types that are possible with a system.

In text IR, there is a almost-complete consensus to choose (sparse) term-document matrices as the basic data type of index data. However, the data structures used differ, ranging from distributed inverted files [TXM03] to the combination of replicated data summaries that allow choosing the most interesting peers [CAN02].

Location of data The diversity of P2P-IR architectures for indexing brings us to the location of data. Most P2P-IR scenarios seem to assume implicitly that each peer holds some local data that is indexed using the P2P system. This scenario is attractive, as it takes replication cost into account as part of the P2P algorithm.

Distinct from these scenarios are scenarios that assume that each peer performs focused web crawling in order to have each peer contain a focused set of documents, focused on its area of interest. This scenario has its advantages, as peer summaries will be more focused and the size of peers can be very homogeneous.

In terms of load balancing it is useful to have peers with similar collection sizes. However, it must be noted, that skewed peer size distributions lead to a simple baseline for peer selection algorithms: In peer selection algorithms, peer data summaries are used in order to choose which peers will receive the query (*e.g.* [GGMT99]). Efficient peer selection algorithms try to minimize the number of peers that need to be visited for processing a query. If the size distribution is sufficiently skewed, a useful selection algorithm is selecting peers by *size*. If we assume every document drawn from the collection to have the same probability of being part of the query result, the number of relevant documents found is proportional to the number of documents present in the selected peers.

In order to give an insight how powerful this method can be, we analyzed images taken from a large `flickr.com` crawl. We model the data collection of a peer as the data collection publically shared by a user account in Flickr. In our 3 million images crawl, 12% of the users held 80% of the images, *i.e.* blindly selecting 12% of the modeled peers would yield 80% of the relevant images searched for by a P2P-CBIR query.

In other words, in the presence of skewed peer data distributions, sophisticated P2P algorithms have to compare to very simple baseline algorithms that already perform well.

4 Cost

There are three types of cost that matter in P2P-IR: *communication cost, storage cost,* and *processing cost.* Each of these types of cost arise in each phase of the life cycle:

Join: A peer joining the network needs to query the network for the right neighbors. Then it has to request its share of indexing data from the network and it has to publish the index data for the documents it holds. The efficiency of the network join will be determined largely by the the caching methods that can be applied, and how expensive the re-activation of cached index data is with respect to the cost of newly publishing the index data. In the case of a crawler-based scenario [NBMW06], before submitting anything, the peer has to crawl or update its crawled collection.

Operation: Much of the current P2P-IR research focuses on the cost of operation [LLH+03, BMWZ05, TXM03]. Distributed indexing structures mostly use the bandwidth of the peers that participate in the query processing. Source selection and unstructured replication based algorithms use both bandwidth and processing power.

Leave: There are two potentially important cost factors. *Firstly,* if a peer leaves, the other peers have to create a new replica of the index data that just left the system [RD01]. The same, it has to assure that the link stays operational. *Secondly,* if a peer leaves, the P2P-IR method has to assure *freshness of index data*. To our knowledge there is little research on how to optimize this.

Most P2P-IR systems focus on the cost of operation. However, in the presence of massive churn, the join and leave costs come into view. Caching index data in order to make joins cheaper seems an attractive idea, but this idea has to be investigated more deeply with given application scenarios in mind.

Similarly, some more care has to be taken in order to guarantee freshness of index data while limiting the cost of expiring index data. Simply periodically re-publishing index data does not seem an option, as even if you re-publish every hour, in many realistic churn scenarios, most of the index data in the system will be stale, *i.e.* pertaining to documents that already left the system.

In all these scenarios we find particularly attractive architectures that distribute the cost of a peer entering the network over time. This might be a viable fashion to cope with cameo peers. Distributed indexing structures tend to force a peer to enter its data into the network at once, when entering the network. Otherwise the data will not be searchable at all. Instead, one could rather enter the index data and adjust the network's connectivity spread over time. In such a scenario, peers that stay just for a short session time cause only a fraction of the cost of a full join/leave cycle. In the beginning, their data is hard to find for other peers and becomes easier to find the longer they stay.

An example for this approach is [TSW04]. It slowly builds up connectivity of peers based on their behavior. However, due to the curse of dimensionality [AHK01, WSB98], it is unlikely that such approaches will work equally well for *e.g.* similarity queries over high-dimensional vectors, as these queries are difficult to route using multi-hop routing.

5 Interaction

While in the network community network latency is an important topic, little attention is given to query processing duration in P2P-IR. In fact, the current development in P2P-IR appears to mimick closely the development in IR itself: Although the interactivity of IR was instrumental to its wide adoption, human-computer interaction properties of IR is only lately becoming a cover topic [WKDS06].

It appears that for interactivity, low query durations are essential. Some of the most famous retrieval applications that couple information visualisation with direct manipulation enable the user to send and to evaluate visually 10 queries per second as reported among others in [SP04]. Evidently, such things would not be possible in current P2P systems.

While we consider that given current networking technology it will be hard (if not impossible) to reach query durations of $0.1s$ per query for full query processing, we consider it a goal worthwile to provide fast approximate query processing (fast *guessing*) in P2P networks that can be the basis for fast feedback. Later in the query process, when most of the index data of the peer is already discarded from consideration, full query processing can be applied to restricted data sets.

Obviously, this problem is hard to solve, as it needs an application and a large fielded network in order to become testable by humans. However, we feel that this perspective is missing from most of the current research in P2P-IR.

6 Conclusion

Within this paper we have identified and described some interesting topics of research in addition to the mostly emphasized query efficiency considerations, namely

Cameo peers: Similar to the classic *free rider*, IR peers have to cope with peers that come just as clients, and not as contributors.

Time of day in regional P2P: Current research focuses on world-wide services with "soft" time of day effects. However, especially when considering non english language P2P-IR, time of day effects might be much stronger than expected.

Skewed data distributions: Current algorithms need to compare to simple methods that simply make use of skewed data distributions that can be found in some usage scenarios.

Fast useful guessing: Currently, the results of P2P-IR algorithms are compared as batch algorithms. However, to be successful P2P needs to support interaction. In order to reach useful turn around times P2P-IR algorithms have to provide fast guessing capabilities that enable the user to see if he is approaching the target of his search or not.

Literatur

[AHK01] Charu C. Aggarwal, Alexander Hinneburg, and Daniel A. Keim. On the Surprising Behavior of Distance Metrics in High Dimensional Spaces. In Jan Van den Bussche and Victor Vianu, editors, *ICDT*, volume 1973 of *Lecture Notes in Computer Science*, pages 420–434. Springer, 2001.

[BMWZ05] Matthias Bender, Sebastian Michel, Gerhard Weikum, and Christian Zimmer. Das MINERVA-Projekt: Datenbankselektion für Peer-to-Peer-Websuche. *Inform., Forsch. Entwickl.*, 20(3):152–166, 2005.

[BSV03] Ranjita Bhagwan, Stefan Savage, and Geoffrey M. Voelker. Understanding Availability. In Kaashoek and Stoica [KS03], pages 256–267.

[CAN02] Francisco Matias Cuenca-Acuna and Thu D. Nguyen. Text-Based Content Search and Retrieval in Ad-hoc P2P Communities. In Enrico Gregori, Ludmila Cherkasova, Gianpaolo Cugola, Fabio Panzieri, and Gian Pietro Picco, editors, *NETWORKING Workshops*, volume 2376 of *Lecture Notes in Computer Science*, pages 220–234. Springer, 2002.

[GGMT99] Luis Gravano, Hector Garcia-Molina, and Anthony Tomasic. GlOSS: Text-Source Discovery over the Internet. *ACM Trans. Database Syst.*, 24(2):229–264, 1999.

[KLVW04] Alexander Klemm, Christoph Lindemann, Mary K. Vernon, and Oliver P. Waldhorst. Characterizing the query behavior in peer-to-peer file sharing systems. In Alfio Lombardo and James F. Kurose, editors, *Internet Measurement Conference*, pages 55–67. ACM, 2004.

[KS03] M. Frans Kaashoek and Ion Stoica, editors. *Peer-to-Peer Systems II, Second International Workshop, IPTPS 2003, Berkeley, CA, USA, February 21-22,2003, Revised Papers*, volume 2735 of *Lecture Notes in Computer Science*. Springer, 2003.

[LLH+03] Jinyang Li, Boon Thau Loo, Joseph M. Hellerstein, M. Frans Kaashoek, David R. Karger, and Robert Morris. On the Feasibility of Peer-to-Peer Web Indexing and Search. In Kaashoek and Stoica [KS03], pages 207–215.

[Men03] Joseph Menn. *all the rave, The Rise and Fall of Shawn Fanning's Napster*. Crown Business, New York, 2003.

[MH03] Wolfgang Müller and Andreas Henrich. Fast retrieval of high-dimensional feature vectors in P2P networks using compact peer data summaries. In Nicu Sebe, Michael S. Lew, and Chabane Djeraba, editors, *Multimedia Information Retrieval*, pages 79–86. ACM, 2003.

[NBMW06] Thomas Neumann, Matthias Bender, Sebastian Michel, and Gerhard Weikum. A Reproducible Benchmark for P2P Retrieval. In Philippe Bonnet and Ioana Manolescu, editors, *ExpDB*, pages 1–8. ACM, 2006.

[RD01] Antony I. T. Rowstron and Peter Druschel. Pastry: Scalable, Decentralized Object Location, and Routing for Large-Scale Peer-to-Peer Systems. In Rachid Guerraoui, editor, *Middleware*, volume 2218 of *Lecture Notes in Computer Science*, pages 329–350. Springer, 2001.

[RFH+01] Sylvia Ratnasamy, Paul Francis, Mark Handley, Richard M. Karp, and Scott Shenker. A scalable content-addressable network. In *SIGCOMM*, pages 161–172, 2001.

[SBR04] Nima Sarshar, P. Oscar Boykin, and Vwani P. Roychowdhury. Percolation Search in Power Law Networks: Making Unstructured Peer-to-Peer Networks Scalable. In *Peer-to-Peer Computing*, pages 2–9. IEEE Computer Society, 2004.

[SMK+01] Ion Stoica, Robert Morris, David R. Karger, M. Frans Kaashoek, and Hari Balakrishnan. Chord: A scalable peer-to-peer lookup service for internet applications. In *SIGCOMM*, pages 149–160, 2001.

[SP04] Ben Shneiderman and Catherine Plaisant. *Designing the User Interface*. Addison-Wesley, 2004.

[TSW04] Christoph Tempich, Steffen Staab, and Adrian Wranik. Remindin': semantic query routing in peer-to-peer networks based on social metaphors. In Stuart I. Feldman, Mike Uretsky, Marc Najork, and Craig E. Wills, editors, *WWW*, pages 640–649. ACM, 2004.

[TXM03] Chunqiang Tang, Zhichen Xu, and Mallik Mahalingam. pSearch: information retrieval in structured overlays. *Computer Communication Review*, 33(1):89–94, 2003.

[WKDS06] Ryen W. White, Bill Kules, Steven M. Drucker, and M.C. Schraefel. Special issue: Supporting Exploratory Search, Introduction. *Commun. ACM*, 49(4):36–39, 2006.

[WSB98] Roger Weber, Hans-Jörg Schek, and Stephen Blott. A Quantitative Analysis and Performance Study for Similarity-Search Methods in High-Dimensional Spaces. In Ashish Gupta, Oded Shmueli, and Jennifer Widom, editors, *VLDB*, pages 194–205. Morgan Kaufmann, 1998.

[YGM03] Beverly Yang and Hector Garcia-Molina. Designing a Super-Peer Network. In Umeshwar Dayal, Krithi Ramamritham, and T. M. Vijayaraman, editors, *ICDE*, pages 49–62. IEEE Computer Society, 2003.