Final report on efficient, decentralized, and scalable techniques for concept-based search, link analysis, and personalized search
Start date of the project: January 2004
Duration: 50 months
Project Coordinator: Prof. Dr. math. Friedhelm Meyer auf der Heide
Heinz Nixdorf Institute, University of Paderborn, Germany
Due date of deliverable: December 2007
Actual submission date: December 2007
Dissemination level: PU – public

Work Package 6.1: Models and Strategies for Collaborative Web Information Search
Participants:
- Max Planck Institute for Computer Science (MPII), Saarbrücken, Germany
- Computer Science Institute (CTI), Patras, Greece
- Telenor Communication AS (Telenor), Oslo, Norway

Authors of deliverable:
- Gerhard Weikum (weikum@mpi-sb.mpg.de)
- Holger Bast (bast@mpi-sb.mpg.de)
- Geoffrey Canright (geoffrey.canright@telenor.com)
- Peter Triantafillou (peter@ceid.upatras.gr)
1 Introduction and Overview

A major goal of DELIS SP6 is to develop foundations for collaborative Web information search in an Internet-scale peer-to-peer (P2P) system. We are aiming at a P2P system where each peer has a full-fledged Web search engine, including a crawler and an index manager. The crawler may be thematically focused or crawl results may be postprocessed so that the local index contents reflects the corresponding user’s interest profile. Each of the peer-specific local search engines could leverage advanced methods such as spectral analysis, linguistic tagging, ontology-based concept search, and other techniques that cannot be applied to a centralized Web-scale corpus. With such a highly specialized and personalized “power search engine” most queries should be executed locally, but the user may occasionally not be satisfied with the local results and would then want to contact other peers. A “good” peer to which the user’s query should be forwarded would have thematically relevant index contents, which could be measured by statistical notions of similarity between peers. Query routing could greatly benefit from collective human inputs in addition to standard statistics about terms, links, etc.: knowing the bookmarks and query logs of thousands of users would be a great resource to build on.

While the first two years of this workpackage have mostly focused on the P2P architecture and collaboration strategies for keyword-search functionality and basic but efficient and scalable query routing, the third and fourth project year have pursued advanced functionality for statistically enhanced query routing, concept-based search, new forms of link analyses for authority and trust assessment, and personalization strategies based on user or community behavior. Last year’s Deliverable D6.1.3 (December 2006) has given a preliminary report on the work in this area. This report - Deliverable D6.1.4 - gives a final account of our results. It extends and supersedes last year’s report; for completeness, results of the third year are also included here.

Our main achievements in this line of research are the following:

• **P2P Query Routing** (mostly work of the 3rd year):
  Our initial work [16] on overlap-aware query routing has been significantly improved for combined quality-novelty in P2P networks, by better estimation of overlap measures among peers and global measures of term specificity like network-wide document frequencies. These results are joint work by MPII and CTI, and have been published in the highly competitive EDBT conference 2006 [90] as well as several refereed and selective workshops [21, 20, 101, 75]. For further improvement of query routing, new ways of detecting and exploiting correlations among different terms have been developed, at local and global scale and for queries as well as peer collections. This joint work by MPII and CTI has been published in the P2P research community’s flagship conference IPTPS 2006 [19] and in the highly selective CIKM conference 2006 [91]. The latter paper won the CIKM 2006 Best Interdisciplinary Paper Award (the conference’s best-paper award, as its mission is to foster integration of DB and IR methods), out of 518 submitted and 81 accepted full papers.
  The entire work on P2P query routing is well documented in full detail in the dissertation [23], completed in July 2007 and graded with the predicate “with distinction”.

• **Concept-based Search** (mostly work of the 4th year):
  MPII obtained new fundamental insights into understanding and explaining LSI and related spectral analysis methods, which in turn led to an algorithm that can effectively extract synonymy, hyponymy, and hyponymy relations from a term-document matrix in a completely unsupervised manner [7, 6].
MPII also developed new approaches for harvesting facts (entities and binary relations) from knowledge sources like Wikipedia, WordNet, and other high-quality Internet sites, leading to the very large and highly accurate ontology YAGO [114]. The methods for the knowledge extraction and integration include both rule-based approaches [115] as well as techniques based on natural-language parsing and statistical learning [113]. A new methodology for searching and ranking on this kind of knowledge base has been developed and prototyped in the NAGA search engine [69, 70].

Both kinds of semantic knowledge - derived from the new interpretation of LSI and from harvesting facts on Internet sources - have been integrated into the CompleteSearch engine developed at MPII for faceted search and entity search in combination with very fast text search. A salient feature of this engine is its novel algorithms for context-aware prefix queries that are used in a highly versatile and efficient manner [8, 9, 12].

Much of the work on concept-based search is described in great detail in the dissertations [85, 126], both completed in November 2007; [126] obtained the rarely given predicate “with distinction”.

- **Top-k Query Processing** (both 3rd and 4th year):
  MPII has developed some of the world-leading algorithms for efficient top-k query processing, based on judicious scheduling of sequential and random accesses to indexes. These results have been published in most prestigious and highly competitive conferences like VLDB and SIGIR [120, 121, 10, 123], and they are included also in a comprehensive article in the VLDB Journal [124].
  For the distributed case with index lists spread over multiple peers, novel algorithms have been developed, with extensive experimental studies as well as mathematical cost models [89, 92]. Moreover, a framework and efficient methods have been developed for optimizing top-k operator trees for execution in large-scale networks [97, 94].
  The different results on top-k query processing are documented in great detail in the dissertations [122, 93, 85]; the thesis [122] has won the Otto-Hahn Medal of the Max-Planck Society and obtained an Honorable Mention (i.e., was among the top 3) for the ACM SIGMOD Dissertation Award 2007.

- **Distributed Link Analysis** (both 3rd and 4th year):
  The work on distributed link analysis with asynchronous collaboration among peers has further matured and established itself among the world-leading methods on this important topic. This includes the joint work by Telenor and UniBo on distributed methods for a wide range of spectral analysis tasks, which resulted in the publications [35, 64], as well as the work on the JXP method for P2P computation of PageRank-style authority measures. The two approaches are complementary by covering different computational models and application areas. The JXP work has partly been performed jointly by MPII and URDLs, and led to very prestigious publications [100, 102, 106] including a VLDB 2006 paper and a comprehensive article in a special best-of-VLDB’06 issue of the VLDB Journal. Recently, additional enhancements for resilience to cheating peers and for applications in analyzing social networks have been developed in [104, 105].

- **Personalized Search and Ranking** (both 3rd and 4th year):
  New methods have been developed at MPII for leveraging query logs and click streams in order to personalize search result ranking, by means of the QRank method, which is based on a Markov reward model and can incorporate also negative assessments, both at the level of individual users and entire user communities [81, 82]. For individual users, a powerful framework has been developed that can improve search results in a variety of situations [50]: recognizing recurrent information needs, establishing and harnessing user background information in advanced search
sessions, and re-ranking search results based on the user’s history of queries, result clicks, and general browsing behavior. Finally, new methods for mining query logs based on IP-level Internet traffic measurements, have been jointly developed by TUM and MPII and published in the highly reputable IMC conference 2006 [66].

2 Main Results

2.1 Query Routing

2.1.1 State of the Art

For P2P search over Web data with ranked retrieval, we are developing the Minerva system [17, 22]. This is one of several projects world-wide (e.g., [1, 43, 107, 116]; see also the book [112] and the references given there) that are pursuing ways of making search-engine technology decentralized, so as to make it ultra-scalable and self-organizing, exploit advanced linguistic and statistical learning models, benefit from individual-user and social-community inputs, and make Web search resilient to manipulation or censorship.

In Minerva, each peer has a full-fledged Web search engine, including a crawler and an index manager. The crawler may be thematically focused or crawl results may be postprocessed so that the local index contents reflects the corresponding user’s interest profile. For collaborative search, the peers are connected by an overlay network based on a distributed hash table (DHT). The DHT also forms the basis of a conceptually global but physically decentralized and scalable directory that contains metadata and statistics about the peers’ contents and quality. Note that, for scalability, the directory is not designed as a page-granularity global Web index, but is limited in size to the number of indexed features (e.g., keywords or topics) times the number of peers. This avoids the pitfalls outlined in [73]. Also note that the pursued P2P Web search includes ranked retrieval and is thus fundamentally much more difficult than Gnutella-style file sharing or simple key lookups via DHTs.

With a user’s highly specialized and personalized “power search engine” most queries should be executed locally, but once in a while the user may not be satisfied with the local results and would then want to contact other peers. This is the query routing (or peer-selection) problem, the cornerstone of the P2P search engine. Although the problem is related to earlier work on metasearch engines and distributed information retrieval [34, 87], the P2P setting is much more challenging because of larger scale and high dynamics. A “good” peer to which the user’s query should be forwarded would have thematically relevant index contents, which could be measured by statistical notions of similarity between peers. To this end, state-of-the-art methods employ term-frequency-based models [34], statistical language models [78, 79], or cost-based decision models [98, 99]. On the other hand, each additional target peer for a query should yield novel results that are not yet provided by previously selected peers or even the local index of the query initiator itself. To this end, we have developed an overlap-aware query routing strategy that combines content-quality measures with Bloom-filter-based estimates of content overlaps among peers [16].

2.1.2 Statistically Enhanced Query Routing

Our initial work [16] on overlap-aware query routing has been further extended and significantly enhanced in [90]. Our method aims to optimize a weighted combination of search result quality and novelty. As the query routing decision made for execution planning is in the critical path of user-perceived response time, fast estimation of quality-novelty measures is crucial. We have developed new methods for this purpose, utilizing compact synopses like Bloom filters [28], hash sketches [55] and min-wise independent permutations [31] in combination with the underlying DHT.

The above considerations show that distributed management of statistical information about peers and their data collections is a key issue. This involves efficient gathering and dissemination of statistics
as well as estimations for specific purposes. A global measure of particular interest (for query routing and query result merging) is the document frequency (df) of a keyword, i.e., the total number of distinct documents in the entire network that contain the keyword. Estimating this number is difficult because of duplicate documents at different peers. We have developed a highly efficient and accurate method for this problem, by combining hash sketches and the DHT-based overlay network [21].

An important network-wide statistical estimation problem is to efficiently determine pairs of highly correlated or anti-correlated keywords, either in queries or in the data. In [19, 91] we have developed a technique that utilizes the DHT-based infrastructure for an efficient solution, which can piggyback all necessary message exchanges on the network traffic that is needed for standard query routing and execution anyway. Awareness of keyword correlations is very useful for more effective query routing decisions.

The problem that mandates correlation awareness is illustrated by the following example. Consider two- or three-keyword queries such as “Anna Kournikova”, “native American music”, or “PhD admission”. A standard approach would decompose each query into individual keywords such as “native”, “American”, and “music”, identify the best peers for each of the keywords separately, and finally combine them (e.g., by intersection or some form of aggregating the summary scores) in order to derive a candidate list of peers to which the query should be forwarded. This approach may lead to mediocre query results as the best peers for the entire query may not be among the top candidates for the individual keywords. In a worst case scenario, these peers might not have a single data item that matches all keywords at once. Hence, we miss out on the fact that, for example, “PhD” and “admission” are statistically correlated in the underlying corpora and that the best matches for the entire query should exhibit a higher-than-average frequency of both keywords.

We have developed two alternative approaches to this problem: sk-STAT, using the already existing single-keyword statistics to estimate a peer’s quality for key sets, and mk-STAT, enhancing the distributed directory that guides the query routing decisions to explicitly include also statistical information about judiciously chosen sets of multiple keys. Our methods can be used with a large variety of P2P overlay networks, including DHTs but also arbitrary graph topologies with requests being routed among peers based on peer-local routing indices. In the DHT case, the statistical information that drives our query routing covers While mk-STAT in principle is the more powerful method, it faces the necessity to identify those valuable key sets that are most likely to enable improvements, as it is practically infeasible to build and disseminate statistics for all possible key sets for combinatorial complexity. Instead, the discovery of interesting key sets is initiated by mining locally gathered query logs, to improve the performance of frequently queried key combinations. This discovery phase can optionally trigger an in-depth statistical analysis of the correlations within the peers’ data collections. One of the paper’s novel key contributions is how to make this analysis efficient and scalable. The approach is highly scalable by piggybacking all network communication for gathering and disseminating statistical information on messages that need to be sent between peers anyway (for their regular query traffic). We employ hash sketches (HS) [55] as compact synopses for capturing key- and key-set-specific collection quality, that we combine efficiently for different keys and from different peers in a distributed setting. The information gained is harnessed by the query routing process, utilizing the DHT infrastructure for efficiency, and leads to significantly better peer selection decisions for subsequent queries.

Query routing makes a peer-selection decision when it sees a query, so it contributes to the user-perceived query run-time. This limits the efforts that query routing can afford. For example, it is out of the question to perform extensive computations for comparing statistical language models of different peers. On the other hand, this consideration motivates pre-computing some of the query-routing decisions and encoding them in a “semantic overlay network” (SON) [42, 133]. ¹ The SON is

¹SON has become an established technical term in the literature. Perhaps, a better characterization would be “statistical overlay network” or “social overlay network".
an additional overlay network that connects frequently and positively interacting peers as immediate neighbors. For example, a peer that often provides good answers to the queries posed by some other peer should become that other peer’s neighbor. This precomputed SON topology can accelerate query routing by preferring SON neighbors. However, the SON does not necessarily pre-determine the query routing decisions. It should still be possible that a peer explores alternatives to its SON neighbors for a particularly important and difficult query. Also, the SON topology itself should, of course, be dynamically maintained as the data, workload, and behavior of the peers in the network evolve over time.

In our recent work [75] and [103] we have developed new kinds of SON architectures, one based on comparing statistical language models between peers and the other based on peer affinity in terms of the Web pages and links known at different peers. The first approach [75] uses the Kullback-Leibler (KL) divergence between the term-frequency distributions of two peers as a measure of peer similarity and aims to connect the most similar peers as SON neighbors. This is a very natural model, as it results from the standard multinomial generative model for statistical language models. However, it faces the difficulty that the KL divergence is not a metric, so there are no obvious ways of avoiding a comparison of all peer pairs for deriving the best SON topology. We exploit recent results from information theory [51] and embed peers into a metric space with appropriate pruning capabilities.

The second approach [103] aims to connect peers such that they are effective neighbors for exchanging complementary information about the Web graph. More specifically, a peer \( y \) is viewed as a “good” neighbor of peer \( x \) if \( y \) has many Web pages that point to pages that are held by \( x \). This approach has immediate benefits for the decentralized link analysis method JXP that will be explained in Subsection 2.4.2. We speed up the comparison of peers by encoding Web pages and the outgoing neighbors of Web pages into synopses based on min-wise independent permutations [31].

### 2.2 Concept-based Search

#### 2.2.1 State of the Art

Lifting search from keywords to semantics concepts and relations can be pursued in various ways. Latent-semantics methods detect dependencies and strong correlations between terms, by spectral decomposition or unsupervised clustering on the term-document matrix of the underlying corpus. A certain degree of information reduction is desired for eliminating noise; this is achieved by mapping the original data onto a lower-rank matrix. LSI [47, 5], pLSI [61, 62], LDA [27], and some forms of statistical language models with background models [77] fall into this category of fully automated methods. Queries are also mapped into the lower-rank space whose dimensions correspond to latent concepts. Search result quality can often be improved this way, but the scalability of these methods is limited, they often require careful tuning of the rank of the latent-concept space, and the quality improvements highly depend on the data and query characteristics, with gains ranging from significant to very marginal.

Explicit knowledge about concepts and semantic relations between concepts is a major alternative, but it requires human work to construct the corresponding knowledge base. Existing thesauri and ontologies such as WordNet [54] can be used to build or at least bootstrap the concept space. This approach yields explicit relations like synonymy, polysemy, hypernymy/hyponymy (aka Is-A relation), holonymy/meronymy (aka Part-Of relation), etc. Co-occurrence statistics gathered on large corpora like Wikipedia or the Web can be used to quantify the “strength” of relations between different concepts. This empowers a search engine to employ various forms of query expansion: adding synonyms to the keywords in a query, adding hyponyms or other strongly related concepts to a query with low recall, or even disambiguating keywords that can have multiple meanings. For these techniques, additional information sources such as relevance feedback, query logs, and click streams can be incorporated and combined with knowledge about semantic concepts. Query expansion has been
shown to boost the result quality of difficult queries, but it is also known as a very subtle technique that may possibly lead to result degradation when applied without sufficient care and fine-tuning [24]. Our work [121] has developed a novel, self-tuning way of making query expansion more robust and efficient by adding expansion terms incrementally at query run-time on demand.

Methods that leverage explicit knowledge can only be as good as the quality of their underlying knowledge base. WordNet, as the most prominent example, is very rich and fine-grained in modeling synonymy, polysemy, and hyponymy of abstract concepts such as “corruption”, “capital”, “state”, or “process”. But it is sparse on concrete entities such as geographic locations and organizations, important people like scientists, contemporary politicians, sports and movie stars, music bands, etc. To obtain such knowledge and also relationships between these entities, information extraction (IE) methods can be applied to text documents, Web forms and tables, and other already existing sources. IE combines pattern matching with linguistic analyses, statistical learning, and lookups in existing lexicons (e.g., so-called gazetteers with city names). The results may include false positives, but the accuracy of these methods is often acceptable, depending on the difficulty of the extraction task. Thus, such automatically gathered concepts and relations can be added into a knowledge base and tagged with their corresponding confidence values. Recent work on IE at a larger scale is reported in [4, 52]; broader surveys include [40, 45].

2.2.2 Unsupervised and Almost-Unsupervised Extraction of Semantic Relations

In our work [7] we developed a new framework for explaining how spectral decomposition and particularly LSI achieves its latent-concept behavior. This framework characterizes the relatedness of two terms by varying the rank of the LSI target space and studying the term-term similarity when transformed into the low-rank space. Different shapes of the relatedness curves, as functions of the rank, allow us to identify, with high confidence, synonyms and other pairs of highly related terms. This enables techniques for document expansion so as to virtually include, with lower weight, such related terms in a document although the term does not literally appear.

The recent paper [6] generalizes our prior work by showing that the spectral decomposition can be used to extract even a term taxonomy with hierarchical hyponymy/hypernymy relations in an unsupervised manner. A mathematical criterion is developed for distinguishing between between four kinds of term-term relations in a given document collection: unrelated (e.g., car – fruit), symmetrically related (e.g., car – automobile), asymmetrically related with the first term being more specific than the second (e.g., banana – fruit), and asymmetrically related with the first term generalizing the second (e.g., fruit – banana). The method has been tested on the ODP (open directory project) corpus, and has been able to reconstruct the OPD hierarchy of category names without any training input.

2.2.3 Concept and Fact Harvesting from High-Quality Knowledge Sources

An alternative way towards concept-based semantic search capabilities is to exploit explicitly represented knowledge in the form of ontologies and thesauri. The latter can be constructed, to some extent, from hand-crafted knowledge such as WordNet, but a larger-scale and self-maintaining, promising approach is to automatically learn the concepts and relations for an ontology directly from rich text corpora such as Wikipedia and other Internet sources. This requires a combination of extraction rules, pattern matching, linguistic analyses, and statistical learning to identify, for example, person names or locations and to extract instances of binary relations such as located-at (city, river), born-in (person, place), plays-instrument (person, instrument), or the generic is-instance-of (entity, concept).

We have developed a comprehensive suite of methods to harvest facts from Wikipedia and integrate them with the higher-level concepts derived the WordNet thesaurus, in order to build a high-coverage
knowledge base with very high accuracy like in hand-crafted ontologies [114]. With this methodology, we have automatically constructed the YAGO ontology (Yet Another Great Ontology) which comprises entities and relations and currently contains more than 1.7 million entities and 15 million facts (instances of relations). These include the taxonomic Is-A hierarchy as well as semantic relations between entities. The facts for YAGO have been extracted from the category system and the infoboxes of Wikipedia, using an extensive system of extraction rules, and have been combined with taxonomic relations from WordNet, using linguistic techniques and heuristics. Type checking techniques help keeping YAGO’s precision at 95% — as shown by an extensive evaluation study [115].

For querying YAGO and similar kinds of knowledge bases, we have developed a semantic search engine coined NAGA (Not Another Google Answer) [69, 70]. NAGA has a graph-based query language geared for expressing queries over entities and relations. Its scoring and ranking model is based on principles of generative language models, and formalizes several notions such as confidence, informativeness and compactness of answers. NAGA has been intensively tested on the YAGO knowledge base, with comparison to other approaches for entity search as well as natural-language question answering.

For extracting further facts from arbitrary Internet sources we pursue a novel approach based on a link-grammar representation of natural-language sentences and an SVM-based statistical learner for determining robust, generalizable linguistic patterns. This approach is implemented in the LEILA prototype system [113]. The method is almost unsupervised by starting with merely a small set of user-provided positive examples and then automatically finding candidate patterns. These patterns cannot be directly applied as they do not generalize well and would lead to many false positives. Instead, we run a dependency parser on the individual sentences, compute a characteristic feature vector from the resulting graph representation of each sentence, and feed these vectors into an SVM classifier.

### 2.2.4 Advanced Methods for Automatic Topic Classification

Another building block in many of these information-extraction and knowledge-harvesting tasks is the automatic classification of text documents or shorter text passages (e.g., for disambiguating entities in textual contexts). We have investigated richer feature models and contextual features of training samples so as to improve classification accuracy with relatively small training sets. One approach is to consider neighbors of text documents in environments with many cross-references such as Web links (but covering also settings such as book, music, or blog recommendations or citation graphs for publications). The graph-based classifier presented in [3] builds on the theory of Markov Random Fields, and additionally develops new techniques for enhanced robustness. This method outperforms purely text-based classifiers and also all previously proposed link-aware classifiers. Finally, we investigate new models that map text features onto semantic concepts in an ontology and leverage the concepts as additional information. In [63] we have developed a generative probabilistic model that uses concepts as a latent-variable layer and an efficient EM-based parameter-estimation procedure. In contrast to previous work on latent semantic models, concepts are explicit and directly interpretable by humans and greatly simplify the model-selection problem of choosing the right number of latent dimensions. In combination with a transductive learning procedure that leverages the richer feature space of unlabeled background corpora, we have built a highly effective classifier that excels especially when training data is very scarce.

### 2.2.5 Efficient Indexing for Auto-Completion and Concept-Enabled Queries

We have developed novel indexing techniques, coined AutoTree and HYB [12, 126], in order to support the automatic completion of query prefixes (both prefixes of individual query keywords and the first few keywords of an incomplete query) given the context of a result candidate set (for previously
evaluated keyword matchings). AutoTree is based on clever bitvector encodings and an appropriately designed index tree. HYB extends inverted files by grouping ranges of keywords and storing postings for all words in the same range in document id order with suitable compression. These techniques allow the query processing for a multi-keyword query to quickly identify possible continuations of a query after having seen a prefix of the query's keywords. Both data structures are highly efficient in terms of both space and time. In practice, HYB outperforms AutoTree by a substantial margin.

The new index structures are highly versatile and provide efficient support for advanced functionalities like phrase matching, faceted search on attribute-value pairs that are associated with text or Web documents (e.g., product categories or conference series for scientific publications), and also concept-based search that considers synonymy, hyponymy, and other semantic relations among keywords and entities [12, 11]. The remarkable point about this is that the richer functionality is achieved by the very same core mechanism that provides the context-aware prefix search. The extended functionalities are implemented in a comprehensive software system, the CompleteSearch engine developed at MPII, and demonstrated with a variety of applications, including faceted search for DBLP and Wikipedia [11, 12, 126].

2.3 Top- \( k \) Query Processing

2.3.1 State of the Art

For Web search, ranking query results by their estimated relevance (according to an appropriate scoring model) is a standard technique for satisfactory user experience. It is also an efficiency issue: the query processor only needs to compute the \( k \) best results (e.g., with \( k \) set to 10 or 100) ideally without having to scan all candidate results that match the query keywords (i.e., in sublinear time).

Such top-\( k \) queries operate on index lists for a query's elementary conditions and aggregate scores for result candidates. One of the best implementation methods in this setting is the family of threshold algorithms (TA) [53, 57, 96], which has been shown to be asymptotically instance-optimal when the number of index lists is viewed as a constant. TA methods aim to terminate the index scans as early as possible based on lower and upper bounds for the final scores of result candidates. Advanced methods from this family [38, 48, 58, 74, 86, 95, 120] perform sequential disk accesses for sorted index scans, but also have the option of performing random accesses to resolve score uncertainty.

When lifting Web search from keywords to concepts, top-\( k \) queries may also become more complex and thus pose efficiency challenges. For example, when we use query expansion based on explicit concept spaces, queries have to operate over a much larger set of index lists; when using document expansion, the average length of index lists increases significantly. These complications do not necessarily affect the asymptotic time complexity of TA, but they do cause major differences in practical run-times. Our methods outlined below have addressed these performance issues.

2.3.2 Index-Access Optimized Algorithms

Advanced TA-style algorithms require scheduling for two kinds of accesses: 1) the prioritization of different index lists in the sequential accesses, and 2) the decision on when to perform random accesses and for which candidates. The prior literature has studied some of these scheduling issues, but only for each of the two access types in isolation [38, 86]. Our recent work [10, 85, 124] has taken an integrated view of the scheduling issues and developed a novel method, coined IO-Top-\( k \) (index-access optimized top-\( k \) algorithm), that outperform prior methods by a large margin. Our main contributions are new, principled, scheduling methods based on a Knapsack-related optimization for sequential accesses and a cost model for random accesses. The methods can be further boosted by harnessing probabilistic estimators for scores, selectivities, and index list correlations.

Compared to a straightforward implementation of the best TA variants, our algorithms and implementation techniques – priority queue management, judicious scheduling, and probabilistic pruning –
yield a performance improvement of two to three orders of magnitude. For example, keyword queries like "kyrgyzstan united states relation" on the 25-million-documents TREC Terabyte corpus can be answered in 10 milliseconds and need to scan only 2 percent of the entries of the 4 relevant index lists. In extensive performance experiments with a variety of large datasets we have often observed that our method is at most 25 percent above a lower bound for the execution cost of any top-k algorithm; so IO-Top-k is empirically near-optimal.

2.3.3 Distributed Top-k Queries and their Optimization

In many applications, the index lists naturally reside on different computers. Such distributed settings, like peer-to-peer systems or sensor networks, pose additional challenges that are much less understood than the centralized problem. In addition to minimizing the computational cost of list accesses, it is crucial to consider network latency and bandwidth consumption for practically viable, efficient solutions. The best pre-DELIS work for this setting was the TPUT algorithm [36]. TPUT operates on \( m \) distributed lists in three phases: the first phase retrieves the best \( k \) candidates from each list and aggregates them; the second phase then generates range queries on all lists to retrieve all further candidates whose local values exceed or equal \( 1/m \) of the rank-k partially aggregated value from the first phase; the third phase performs random lookups to complete the aggregation for all retrieved candidates and safely determine the top-k result items. We have developed new algorithms, analytic cost estimates, and optimization techniques that achieve major performance gains over TPUT.

Our contributions include the KLEE family of algorithms [89, 92, 93] which generalizes and improves TPUT by adding statistical means for predicting scores of candidates after the first phase of initial retrieval. To this end, index lists are enhanced by score-distribution histograms and a set of Bloom filters that capture the items whose scores fall specific score intervals. The query coordinator that is responsible for aggregating partial scores harnesses these synopses for deriving a higher threshold for the second phase’s range queries and saving substantial execution and communication costs in this most expensive step. This approach leads to an approximation algorithm, but the error is typically negligible with provable probabilistic guarantees. If desired, KLEE can also be turned into an exact algorithm by an additional step.

For the approximation guarantee and for tuning KLEE to application needs, an analytic cost model for distributed top-k queries has been developed. The model estimates the depth of the phase-two range queries and the expected number of candidates that determine the networking and processing costs; it yields fairly accurate estimates and can be efficiently evaluated. Its elegant use of principled statistics and its prediction accuracy make this approach far superior to previously proposed cost models for top-k querying.

In addition, several important optimizations have been developed for top-k queries that involve several tens or even hundreds of input lists [97, 94]. The first optimization groups input lists and aggregation operators in a hierarchical way to form execution trees, and uses dynamic programming over the cost model to derive the best plan. The second optimization allows the second phase of KLEE to generate range queries with different value threshold on different lists, by utilizing the histogram statistics that are retrieved in the first phase. The third optimization allows sampling a subset of the input lists rather than scanning all of them, and it chooses the sampled lists in an intelligent manner again utilizing histograms.

2.4 Extended Link Analyses

2.4.1 State of the Art

Authority analysis on link graphs typically builds on spectral methods, i.e., computing Eigenvalues and Eigenvectors and characterizing the corresponding decomposition of a graph’s adjacency matrix. PageRank (PR) [30] and HITS [71] are the seminal methods for link analysis on the Web graph, which
can be used to derive authority scores for Web pages that reflect query-independent importance and community endorsements. [29] and [72] provide good in-depth surveys of follow-up and related work, including their mathematical foundations.

PageRank computes the principal Eigenvector of a matrix that is derived from the Web link graph with an additional feature of uniform random jumps. PageRank vectors, characterizing the prestige of Web pages, are usually computed by the iterative Jacobi method (aka. power iteration) with fast convergence. The same mathematical algorithm has been applied to computing TrustRank [59], SpamRank [15], and other methods of social authority and (positive or negative) impact (e.g., [14, 68, 60]). A general framework for different types of trust and distrust propagation in a graph of Web pages, sites, or other entities is introduced in [56].

A P2P network is a natural habitat for “social search” that leverages community assessments. A simple form of such community input are link analysis methods like PageRank or HITS. But these are centralized algorithms with very high memory demand, and their distributed variants assume that the underlying Web graph can be nicely partitioned among sites. Recently, various distributed versions of link analysis methods have been proposed (e.g., [2, 32, 46, 67, 109, 125, 131]), but they all assume that the global graph is partitioned into disjoint fragments across Web hosts or peers.

2.4.2 Decentralized Methods for Authority and Trust Assessment

In [35, 64, 25, 26] we have developed a comprehensive framework for fully distributed versions of a wide variety of link analysis methods including PageRank but also variants with undirected graphs or un-normalized transition matrices. When propagating authority mass from a node to its successor, our distributed methods use message passing whenever the two connected nodes reside on different computers. These novel methods do not require any global coordination among peers, other than an agreed updating rate. In particular, global synchronization of the individual updates is not needed. The methods have been shown to be robust against high levels of noise, such as message delay, message delay jitter, and message loss. The fine-grained message passing of our approach is appropriate, and an intriguing alternative to centralized link analysis, when peers are willing to collaborate and the underlying network has relatively short latency and very good connectivity.

A different setting arises when we emphasize the autonomy of peers, like in the Minerva prototype where peers can gather their own local content at their discretion. We have developed the JXP algorithm [100, 102, 106] for computing PageRank-style authority scores (PR) of Web pages that are arbitrarily distributed over many sites of a peer-to-peer (P2P) network. Peers are assumed to compile their own data collections, for example, by performing focused Web crawls according to their interest profiles. This way, the Web graph fragments that reside at different peers may overlap and, a priori, peers do not know the relationships between different fragments. The JXP algorithm runs at every peer, and it works by combining locally computed authority scores with information obtained from other peers by means of random meetings among the peers in the network. The computation on the combined input of two peers is based on a Markov-chain state-lumping technique, and can be viewed as an iterative approximation of global authority scores. JXP scales with the number of peers in the network. The computations at each peer are carried out on small graph fragments only, and the storage and memory demands per peer are in the order of the size of the peer’s locally hosted data. It is proven that the JXP scores converge to the true PR scores that one would obtain by a centralized PR computation on the global graph.

The open and anonymous nature of P2P networks opens doors to manipulation of information and quality ratings. For example, in the JXP algorithm we may face cheating peers that aim to distort the global authority values by providing manipulated data in the peer meetings. We have also addressed this issue of misbehaving peers. An extended version of JXP, coined TrustJXP, provides a variety of countermeasures, based on statistical techniques, for detecting suspicious behavior and combining JXP rankings with reputation-based scores [104, 106].
PR-style authority analyses can easily be extended to study the structure of the now popular Web 2.0 applications that exhibit a natural graph structure. These include social networks (e.g., MySpace, FaceBook, etc.) and social tagging platforms (e.g., Flickr, del.icio.us, etc.). Finding the most trustworthy or most important authorities in such communities is a pressing need, given the huge scale, high dynamics, and also anonymity of such social networks. For these reasons, again a decentralized, highly scalable approach seems most desirable. We have started to investigate how the JXP framework can be applied and extended to these settings with more general kinds of spectral analyses over heterogeneous graph structures [105].

2.5 Personalized Search and Ranking

2.5.1 State of the Art

Information about user behavior is a rich source to build on. This includes relatively static properties like bookmarks or embedded hyperlinks pointing to high-quality Web pages, but also dynamic properties inferred from query logs and click streams. For example, suppose a user clicks on a specific subset of the top 10 results returned by a search engine for a query with several keywords, based on having seen the summaries of these pages. This implicit form of relevance feedback establishes a strong correlation between the query and the clicked-on pages. When a user does not click on any of the top 10 results for a given query and rather chooses to rephrase the query using different keywords, this may be interpreted as negative feedback on the relevance or quality of the initial results. Exploiting this kind of user-behavior information can help to improve search result quality for individuals as well as entire communities by adding more “cognitive” elements to the search engine.

The literature contains interesting work in the outlined direction (e.g., [127, 44, 65, 111, 117]), but is still far from a comprehensive and fully convincing solution. [127] applied clustering to query logs in order to identify frequently asked questions regardless of keyword variations. [44] exploited knowledge about similar queries for query expansion. [65] analyzed the statistical significance of interpreting non-clicked top-10 results as negative feedback and found that only non-clicks that are followed by a click-on document at a lower rank should be viewed as relative user preferences. [108] used statistical learning to identify query chains of reformulated queries for the same information demand by the same user. [111, 117] enhanced statistical language models for query-result ranking by background “knowledge” obtained from a user’s long-term query log and click stream.

Link analysis methods such as Google’s PageRank are often based on a random-walk model over the Web graph. Our prior work [80] generalizes the notion of a “random surfer” into a “random expert user” by enhancing the underlying Markov chain to incorporate also query nodes and transitions from queries to query refinements as well as clicked-on documents. Transition probabilities are derived from the statistical analysis of query logs and click streams. In addition, virtual links can be created from textual references using named-entity recognition techniques. The resulting Markov chain converges to stationary authority scores that reflect not only the link structure but also the implicit feedback and collective human input of a search engine’s users. We experimentally showed that this approach could significantly improve the user-perceived quality of search results for advanced queries.

2.5.2 Exploiting User and Community Behavior for Personalized Ranking

Our approach is based on a Markov-chain model with queries as additional nodes, additional edges that capture query refinements and result clicks, and corresponding transition probabilities. Similarly to Google’s PageRank, the computation of stationary visiting probabilities yields the behavior-aware authority ranking. Our original model could not express negative feedback, as probabilities are non-negative and $L_1$-normalized. To capture and exploit also negative assessment such as assigning trust levels to Web pages (e.g., marking a Web page as spam, low-quality, out-of-date, or untrusted), we are pursuing several extended approaches, one of which is based on a Markov reward model [119]
where the assessment part is uncoupled from the random walk in the extended Web graph. A page receives a specific lump reward each time a transition is made to it, and this reward depends on the transition’s source and target and will be derived from the query-log and click-stream information as well as explicit page assessments. The reward itself can be positive or negative. The most interesting measure in such reward models is the expected earned reward per time-step. Results with this approach are presented in [81, 82].

Another line of research that has been explored in [50] is to exploit the user’s history of queries and result clicks for query modification and result re-ranking in a context-aware manner. Prior work did not distinguish the diverse nature of query contexts, and therefore, albeit often leading to result improvements, do sometimes fail and can even lead to undue topic drifts and a degradation of search result quality. Our novel solution distinguishes three kinds of search situations: recurring queries where the user wants to retrieve some information that she has already seen before but does not remember in detail, query refinements within a session that pursue a specific search topic, and queries delving deeper into a topic of user interest. We have developed robust query expansion methods for each of the three cases. Simple but effective heuristics are used to recognize each of the three situations, so that the appropriate actions can be applied.

The evaluation of such personalization models requires experiments with real users’ behavioral traces. To this end, TUM and MPII have jointly developed a novel way of gathering query logs and click streams by monitoring and analyzing network traffic at the IP level. This work has also led to a finite-state transition model that captures the user behavior in query sessions [66].

3 Dissemination of Results

The research in this WP has led to 18 publications [11, 12, 13, 22, 50, 64, 69, 70, 82, 97, 92, 104, 103, 105, 106, 114, 124, 130] in international conferences and refereed workshops in the year 2007 alone (including a few papers that will appear in early 2008). The publication list includes 6 papers in the prestigious, highly selective first-rate conferences WWW, SIGIR, VLDB, CIKM, and ICDE. The journal article on JXP [106] is part of a Best-of-VLDB-Conference special issue of the VLDB Journal, and in 2006 the paper [91] won the best-paper award at CIKM 2006. In addition, the research around this workpackage led to 5 completed dissertations, several of which were graded with the rarely given predicate “with distinction”; one of them has won the Otto-Hahn Medal of the Max-Planck Society and obtained an Honorable Mention for the ACM SIGMOD Dissertation Award. All referenced papers are available on request from the coordinating site UPB.


Many results are also implemented in the Minerva software, a prototype testbed for P2P Web search. Two previous releases of Minerva have been made available as open source software as DELIS deliverables in July 2005 and January 2007; the new Deliverable D.6.5.5 contains an updated - functionally extended and intensively tested - version as of December 2007. Minerva has been demonstrated, with excellent feedback, at three major conferences: an older version at VLDB 2005 in Trondheim and Middleware 2005 in Grenoble [17, 18], and a new, significantly extended, version at the CIDR 2007 conference in Asilomar [22]. Minerva has been used by several DELIS partners, and it is being adapted and extended to the needs in the new FP6 STREP SAPIR on Search for Audio-Visual Data Based on Peer-to-Peer Information Retrieval, which started in January 2007 and involves the DELIS
partners Telenor and MPII.

References


[10] Holger Bast, Debapriyo Majumdar, Ralf Schenkel, Martin Theobald, Gerhard Weikum: IO-Top-k: Index-access Optimized Top-k Query Processing. International Conference on Very Large Data Bases (VLDB), Seoul, Korea, 2006. (DELIS-TR-0324)


[80] Julia Luxenburger, Gerhard Weikum: Query-log based Authority Analysis for Web Information Search, International Conference on Web Information System Engineering (WISE), Brisbane, Australia, 2004


[90] Sebastian Michel, Matthias Bender, Peter Triantafillou, Gerhard Weikum: IQN Routing: Integrating Quality and Novelty for Web Search. Int. Conf. on Extending Database Technology (EDBT), Munich, Germany, 2006. (DELIS-TR-0290)

[92] Sebastian Michel, Thomas Neumann: Search for the Best but Expect the Worst - Distributed Top-k Queries over Decreasing Aggregated Scores. International Workshop on Web and Databases (WebDB), co-located with ACM SIGMOD, Beijing, China, 2007. (DELIS-TR-0603)

[93] Sebastian Michel: Top-k Aggregation Queries in Large-Scale Distributed Systems, Doctoral Dissertation, Max-Planck Institute for Informatics and Saarland University, Saarbruecken, Germany, July 2007.

[94] Sebastian Michel, Thomas Neumann, Matthias Bender, Ralf Schenkel, Peter Triantafillou, Gerhard Weikum: Optimizing Top-k Queries over Distributed Sources, Submitted for Publication.


[113] Fabian Suchanek, Georgiana Ifrim, Gerhard Weikum: Combining Linguistic and Statistical Analysis to Extract Relations from Web Documents, ACM International Conference on Knowledge Discovery and Data Mining (KDD), Philadelphia, USA, 2006. (DELIS-TR-0453)


