Computing and Informatics, Vol. 29, 2010, 571–584

LEARNING PROFILES FOR HETEROGENEOUS DISTRIBUTED INFORMATION SOURCES

Hans Friedrich Witschel

SAP Research CEC Karlsruhe
Vincenz-Priessnitz-Str. 1
76131 Karlsruhe, Germany
e-mail: hans-friedrich.witschel@sap.com

Revised manuscript received 24 March 2010

Abstract. This paper experimentally studies approaches to the problem of describing heterogeneous information sources in distributed environments. In particular, we consider a scenario where a large number of end users can share and retrieve text documents over a peer-to-peer network. Descriptions (or profiles) of peers are useful in a number of applications, such as query routing, overlay construction and expert search. The approach proposed in this paper introduces a new learning method that boosts the weight of query terms in a peer's profile when the peer provides useful documents w.r.t. a given query. Experimental results show high potential for this method. Therefore, various extensions are proposed that involve more user interaction.

Keywords: Peer-to-peer networks, information retrieval, query routing

1 INTRODUCTION

Storing and accessing text-based information via peer-to-peer networks has received increased attention lately due to a number of advantages that it offers over centralised client-server solutions: it offers greater ease of publishing and discovery of resources – since no crawling is necessary – and it significantly reduces maintenance costs and risk of failure because there is no central server that constitutes a single point of failure.

Therefore, we want to consider a scenario where a large number of end users can share and retrieve text documents, ranging from personal notes to official documents, over a peer-to-peer (P2P) network. Peer clients may run on desktop computers, laptops or other mobile devices. Usually, peers that share information are connected to neighbouring peers in a so-called *overlay network*. For performing search, a peer forwards queries to a subset of its neighbours, which will search their local database and then proceed in the same way. The entire process of forwarding user queries in a P2P network is often called *query routing* in peer-to-peer information retrieval (P2PIR).

In this work, we want to consider descriptions of peers – or *profiles* as they will be called later. These are useful in various respects:

- They can support query routing – where forwarding decisions are taken by matching a query against the profiles of neighbours.
- In addition, they can be useful for the construction of semantic overlays – where peers with similar profiles may be clustered.
- As indicated above, we assume that a peer entity corresponds to a single user who has certain interests that are reflected by the documents it shares. Hence, the profile should describe the interests and/or the competence of that person as accurately as possible. If we can achieve that, then profiles become useful also for expert search. That is, we envision that the overlay network of the P2P system and the peers' profiles can not only be used by machines, but can also be interesting for humans to browse and inspect.

Here, we will concentrate on the most common form of profiles in information retrieval scenarios, namely so-called unigram language models, i.e. simple lists of terms with weights. A central question that arises is how to choose the terms that form a profile in order to optimise them both for query routing and usefulness for humans. There are two popular options:

- Content-based: extracting terms from the documents that an information source contains and weighting them according to their presumed importance. The profiles then contain the highest-weighted terms that are extracted in this way. This approach does not take into account whether certain topics that a peer offers are requested frequently or not (at all).
- Query-based: exploiting user interaction and responses to queries. Here, queries are initially routed e.g. randomly; whenever a peer returns a (good) result for a given query, that query or its constituent terms are added to the profile of the peer which is henceforth used in the routing algorithm.

In this work, a combination of both approaches is proposed that starts with content-based profiles and improves them using information from query logs. The adaptation relies on an automatic assessment of the (likely) quality of results returned by peers. The idea is that we want to boost the weight of a query term in a peer's profile if the peer has provided a good answer to the given query. This idea is experimentally compared to using content-based profiles only and to query expansion – the other traditional method of overcoming vocabulary mismatches between queries and profiles. In addition, we present ideas for involving users in the process of profile learning. All in all, the results of this study show that tracking of elementary user interaction (e.g. query logs) is much more effective than query expansion w.r.t. improving query routing.

The rest of this paper is organised as follows: Section 2 summarises the state of the art in the field of profiles and query routing in P2PIR. In Section 3, the approach of profile adaptation is explained in detail. An evaluation environment and experimental results with the proposed algorithms are presented in Section 4, possible extensions of these algorithms – involving more user interaction – in Section 6. Section 7 summarises the findings of this work.

2 RELATED WORK

2.1 Profile Refinement

Peer-to-peer information retrieval (P2PIR) shares many problems with distributed IR (DIR, cf. [3]). In DIR, a central instance, often called *broker*, receives user queries, forwards them to a selection of IR databases and then merges the results returned by these into a final ranking.

The common approach for query routing in DIR treats each information resource as a giant document, creates a profile in the form of a unigram language model from that document and applies conventional retrieval functions in order to rank these profiles and then select the top-ranked ones.

In P2PIR there is no such commonly accepted peer representation. Although the representation of peers by unigram language models is also used [10, 5], a number of alternatives exist, e.g. approaches that use categories from *ontologies* or *taxonomies* to represent peers [2]. The main reason for the emergence of these alternatives is the need for *compactness* in P2PIR: peer profiles often need to be sent around to other peers and stored in their routing tables. Bandwidth and storage limitations present in P2P settings make it necessary for profiles to be very compact.

The basic idea of profile refinement is to characterise a peer or information resource not only by the content that it offers, but by the queries for which it provides relevant documents. Of course, this only starts to work when a significant number of queries have been asked and used as a "training set". It also implies that systems will tend to provide better answers for popular queries, but may fail to do so for unpopular ones.

In P2PIR, many systems use what could be called a *collective* discovery approach to profile refinement by having every peer in the system store query-related information associated with other peers [1, 9]. This results in a resource description (profile) consisting of the past queries that a peer has answered. This profile is, however, not accessible at one central point, but can be thought of as knowledge shared throughout the community. An explicit *learning* approach – taking into account the quality of results returned by a peer – is introduced in [1] where peer profiles are adapted by assigning weights to term-peer (t, p) routing table entries that reflect the estimated quality of the results returned by p w.r.t. queries that contain term t.

2.2 Query Refinement

Another – traditional – way to overcome the mismatch problem between queries and profiles is to refine the queries instead of the profiles, e.g. by query expansion. As pointed out in [19], query expansion is beneficial because it makes up for the loss of document boundaries in profiles: for a phrasal query like "white house", there will probably be many information resources that have both terms in their profile. But how many of their documents will actually contain both terms, let alone the complete phrase?

Two studies [19, 12] examine the effectiveness of query expansion in distributed information retrieval, reaching rather different conclusions. Xu and Callan [19] find significant improvements over the baseline CORI collection selection (cf. [3]), whereas Ogilvie and Callan [12] carry out similar work, but using smaller collections for expansion and with rather discouraging results.

Query expansion is also used in some approaches to P2PIR: in [4], a local pseudo feedback approach based on language modeling is presented, first ranking peers w.r.t. the unexpanded query and then using the best k results returned by the top-ranked peer for pseudo feedback. A study of pseudo feedback performed on large "external" collections for query expansion in ordinary IR can be found in [6], where several very large collections are used for learning a relevance model.

2.3 Contribution

So far, many advanced solutions to peer (or collection) selection and query routing have been proposed and most of them have been evaluated in isolation. There have been comparative evaluations in distributed information retrieval (DIR), e.g. [7], but, to the best of my knowledge, this is the first evaluation that compares a selection of approaches against each other in a unified P2PIR evaluation setting, including methods for profile adaptation that have not been explored in DIR.

3 PROFILE ADAPTATION

This section describes a new approach to learning better profiles that is based on the idea of boosting the weight of a query term in a peer's profile if the peer has provided a good answer to the given query.

We assume that every peer in our P2P system has an initial unigram language model as a profile that is computed from the documents that the peer shares. Because of the need for compactness of peer profiles mentioned earlier, the profile that is stored in other peers' routing tables may only contain the terms with the highest weights, i.e. the profile may have to be pruned. We assume, however, that the peer itself keeps a full profile with all terms in it. Thus, adapting weights of terms in that full profile may result in a different term ranking and thus in a different selection of terms for the pruned profile.

The actual adaptation of profiles relies on a simple learning rule inspired by the reinforcement learning in [18, 1]. The idea behind that approach is to boost weights of query terms in a peer p's profile if p has high-quality results for the query. In [1], the quality of p's results is measured via the average of the scores of contributed documents. Since these scores may not be comparable across peers, we use another measure instead of the average scores, namely RP (relative precision), which is introduced in [17]. More precisely, when an answer to a query $q = (t_1, \ldots, t_n)$ is received from peer p, the weight $w_{i,p}$ of query term t_i in p's profile is updated via

$$w_{i,p}(t+1) = \left(\frac{RP@k(D_p, D_o) + 1}{\text{AVGRP} + 1} \right) w_{i,p}(t). \quad (1)$$

Here, D_p is the result list returned by peer p, D_o is the result list returned by all other peers the query has reached. AVGRP is the average over all RP values of those peers. In the experiments below, $k = 10$ was used throughout. For now, it is sufficient to know that RP measures how highly (on average) the results in D_p are ranked in D_o. Hence, it is a measure of the quality of the results returned by peer p that is solely based on the *ranks* of those result documents in a reference ranking D_o.

As an example, consider the query "white house" and a peer p returning a ranking $D_p = [d_1, d_2]$ of two documents. Now, p learns of the results D_o of all other peers that have contributed to the query; based on this knowledge, p computes $RP@k(D_p, D_o)$ as a measure of quality of its own results, as well as the average RP value $AVGRP$ taken over all contributing peers' results. Now, if $RP@k(D_p, D_o)$ is greater than $AVGRP$, p will increase the weight of the terms "white" and "house" in its profile as prescribed by Equation (1).

In practice, the learning is performed on a query log. This query log is partitioned into a training and a test set of queries. During training, we assume – optimistically and merely for the purpose of evaluation – that each training query reaches *all* peers and that hence D_o consists of all documents found by a centralised system.

For each peer p that possesses at least one document $d \in D_o$, we compute the new weight of query terms in p's profile as given in Equation (1). The update of $w_{i,p}$, however, is only executed if the ratio $\frac{RP@k(D_p,D_o)+1}{\text{AVGRP}+1}$ is greater than 1.

Note that in a real P2P system, when peers manage their own profiles, this procedure requires that peers on the routing path are able to compare their results to that of the others. This can be achieved e.g. by the querying peer – having received the results – computing scores for peers and notifying those with a ratio of $\frac{RP@k(D_p,D_o)+1}{\text{AVGRP}+1}$ greater than 1.

Since the weights $w_{i,p}$ may grow exponentially large with this approach, the final weights $w'_{i,p}$ in peers' profiles are rescaled with a logarithm: $w'_{i,p} = \log(1 + w_{i,p})$.

This way of rescaling was found to work best in a preliminary set of experiments. After training is completed, query routing is performed by matching queries from the test set against the adapted profiles. The test set is identical to the queries used to evaluate all other strategies (that is, baseline and query expansion methods).

4 EXPERIMENTAL SETUP

4.1 Evaluation Procedure

Before starting to discuss the framework used to evaluate the new profile adaptation approach, I will introduce a few choices of parameters that were fixed in the experiments. We assume that each peer truthfully creates and manages its own profile, which in turn allows for the most important simplification that is being made in this work: in an attempt to study the query routing problem in isolation – independent of overlay topology – we only evaluate a DIR scenario, no real P2PIR simulation is performed. Apart from the wish to decouple neighbour selection and query routing, this decision is expected to help reduce the number of free parameters considerably: when trying to simulate a P2P community, we need to make assumptions regarding not only the topology of the overlay, but also the distribution of queries among peers, whether or not forwarding to more than one peer is allowed, churn (i.e. whether or not a contacted peer is on-line or not) etc.

However, the claim is made that the results obtained in the experiments below are valid not only for DIR, but also (and even more so) for P2PIR. In fact, by not committing to particular settings of P2PIR parameters, we can expect the results obtained to be valid across a large number of P2PIR systems with very different settings of these parameters.

On the one hand, the above claim is based on the assumption that a query routing algorithm that performs well in a situation where all peers' profiles are known – i.e. in DIR – will also do so when applied to only a subset of these – as is typically the case in P2PIR. On the other hand, care is taken to design characteristics of the DIR simulation in a way that is typical for P2PIR scenarios – as *opposed* to DIR scenarios. The most important of these characteristics are the following:

- Profiles are pruned (with varying sizes). This is untypical in DIR because there are normally no size restrictions for resource descriptions.
- Peers are expected to be cooperative, i.e. each peer (truthfully) creates its own profile; in DIR, descriptions are most often created by query-based sampling, assuming that collections are uncooperative.
- In DIR, evaluations usually use at most a few hundred information resources, in P2PIR we want to use far more peers, at least a few thousand.
- While information resources in DIR are normally large and semantically heterogenous, peers can be expected to share a smaller amount of documents belonging only to a few selected topics.

The basic procedure applied in all evaluations of this section is to judge the quality of a peer ranking w.r.t. a query by the quality of the results that will be retrieved if peers are visited in the order implied by the ranking.

As described in [15], local idf values are estimated using a mixture of the British National Corpus (BNC) and a sample of 256 documents from the CiteSeer collection. Since all peers hence use the same idf estimates, document scores are comparable across all peers. Thus, merging rankings is trivial: when visiting peer i, its set of documents is united with the documents found at peers $1, \ldots, i-1$ and the resulting set of documents is sorted by the documents' global scores and pruned to a length of 1 000.

4.2 Test Collection

For the experiments that were performed to explore the effectiveness of the new profile learning approach in terms of query routing quality, a scenario was chosen in which individuals within a research community share their own publications. The motivation for applying a P2P solution in this scenario is ease of publishing and topicality. The CiteSeer database of documents was used (containing 570 822 abstracts, written by a total of 230 922 authors), together with an access log consisting of 712 892 successive queries and dating from August and September 2005.[1]

Authors are identified with peers, i.e. a peer shares the documents that the corresponding person has (co-)authored. Because of lacking relevance judgments, the performance of queries from the log is evaluated by comparing the results of a distributed search against those of a centralised system. The measure that was used for this comparison – presented in [16] – is called relative precision (RP) and exploits the ranking of the centralised system as an indicator of probability of relevance. It computes the average probability of relevance of the n top-ranked documents that the distributed system returns.

For the experiments, the last 10 000 queries of the log were used to evaluate all strategies. The first 702 892 queries were used as a training set in the evaluation of profile adaptation. The test set contains 6 883 distinct queries, of which 1 544 occur more than once.

4.3 Preparations

Before applying the new learning method to peer profiles, we need to compute initial, content-based profiles from the documents a peer shares. For that purpose, a variant of the CORI algorithm for ranking information resources (see [3]) was applied. Afterwards, profiles were compressed by simple thresholding applied to the list of terms ranked by CORI weights, i.e. the n terms ranked most highly by their CORI weight will form the profile of peer p. In the experiments, the n-values 10, 20, 40, 80, 160, 320 and 640 are explored and compared to using uncompressed profiles.

[1] Special thanks go to Lee Giles for making the access logs available to me.

The sizes of profiles are absolute and not relative to the size of a peer's collection, because we must assume that the maximum acceptable size of a profile is defined by some technical constraints dictated by the underlying network.

As a baseline for query routing (named "CORI" in the experiments), peers are ranked by the sum of the CORI weights of all the query terms that are contained in the (pruned) profile. Each peer that receives the query retrieves documents from its local repository using the BM25 retrieval function.

Finally, the query expansion methods used in experiments are based on local context analysis (LCA, see [20]). Three types of collections were used to obtain samples of potentially relevant passages:

The web: In that case, queries are passed to the API of a web search engine (Yahoo! in case of the experiments below) and the snippets for the top 10 results are retrieved.

Local pseudo feedback: A local expansion strategy as described in [4] first ranks peers using the original query, retrieves the 10 best results returned by the top-ranked peer and feeds them into LCA.

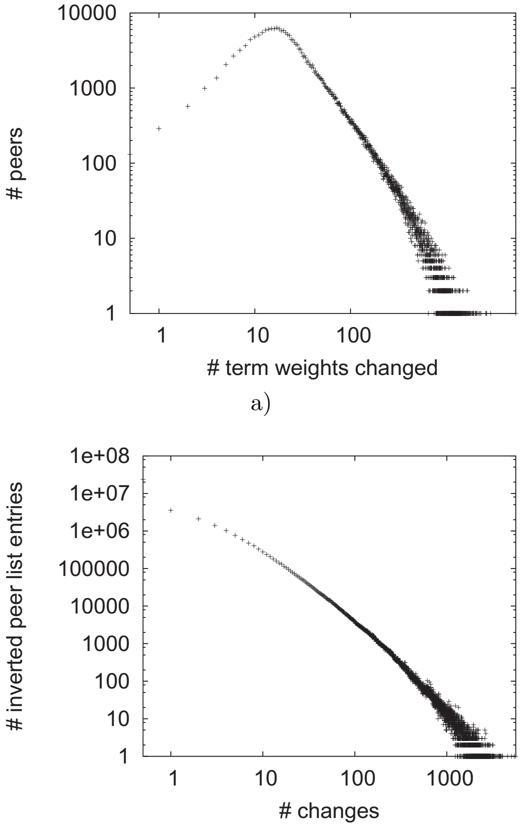
Global pseudo feedback: Instead of using documents only from the top-ranked peer, this strategy assumes knowledge of the whole distributed collection and uses the 10 best results that a centralised system would return as an input to LCA. This strategy cannot be applied in a P2P setting, but it can serve to show how effective pseudo feedback could be if we had complete knowledge.

5 RESULTS

We now turn to the evaluation of the profile adaptation technique presented in Section 3 and described by Equation (1). As mentioned in Section 4.2, the first 702 892 queries of the original CiteSeer query log were used for training and the retrieval with adapted profiles was then performed on the same test set used for baseline and query expansion experiments, consisting of the last 10 000 queries of the log. Updates of profiles were only performed during training, not during the evaluation of queries in the test set. All results are in terms of relative precision at 10 documents (RP@10).

Figure 1 gives information on the number of changes that occur in peer profiles during training. We see that the number of changes applied to individual term-peer pairs (Figure 1 b)) approximately follows a power law, i.e. there are few entries that are updated many times; the vast majority of entries is rarely updated and 62.7 % of the entries are never updated at all. However, only 131 of the 230 922 peers (0.06 %) have none of their term weights changed (Figure 1 a)). This implies that almost all peers have some, but few, of their term weights updated, some of these many times.

All in all, this preliminary analysis shows that profile adaptation can have a considerable impact since there are enough changes during training that can affect the processing of queries in the test set.



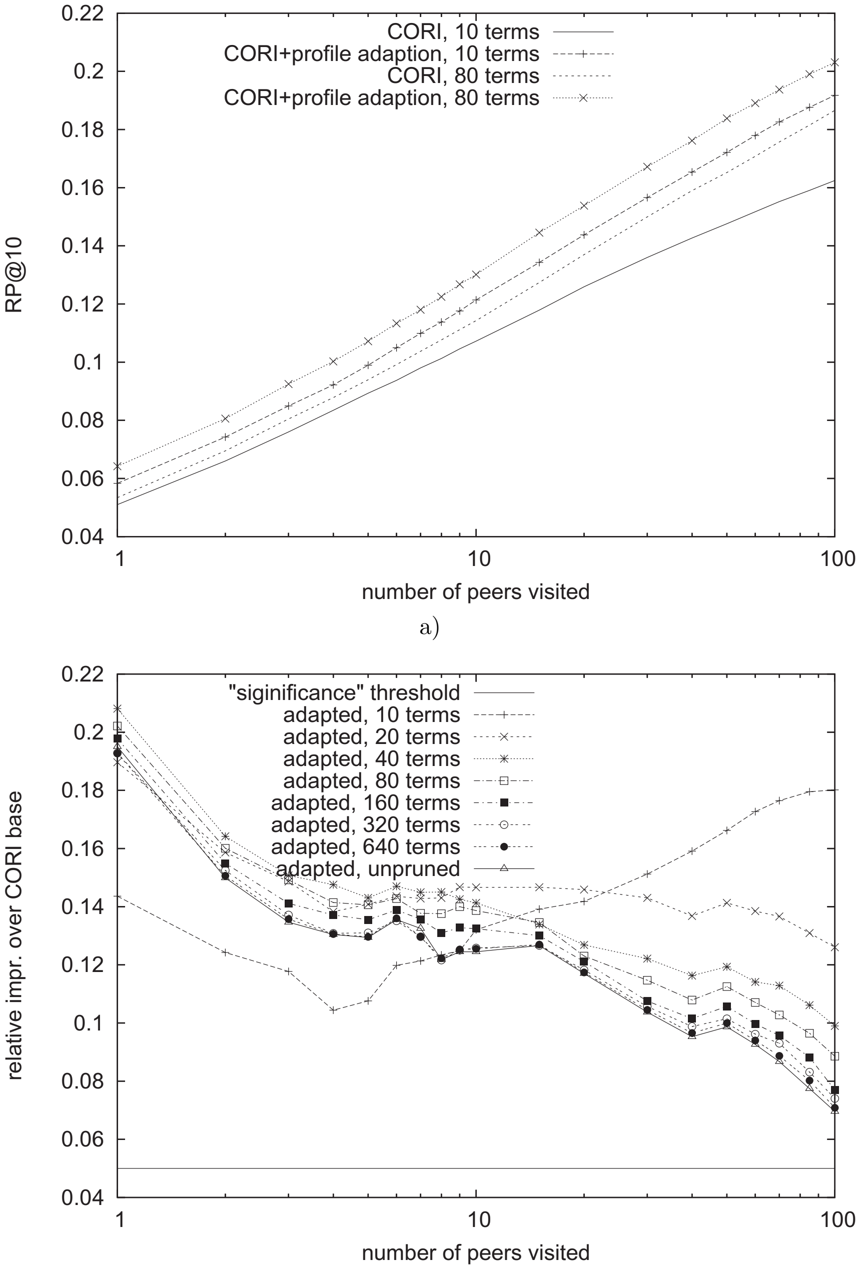
a)

b)

Fig. 1. Histograms of number of changes applied to a) term weights at a given peer and b) a given term entry in a given peer's profile. More precisely, a point at position (x, y) in part a) means that there are y peers, for which exactly x terms had their weight changed. A point (x', y') in part b) signifies that there are y' entries in the entirety of all profiles that were subject to exactly x' changes.

Figure 2 a) shows the performance of CORI baseline runs that use adapted profiles as compared to the CORI baseline with unadapted profiles for profile sizes of 10 and 80 terms. There is improvement for each number of peers visited. Figure 2 b) shows the relative improvement of adapted profiles over unadapted ones as a function of the number of peers visited.

We can see that generally the relative improvement of profile-adapted runs over the baseline is always greater than 5 %. In addition, the relative improvement curves have very similar shapes, with a tendency of smaller profiles gaining more from profile adaptation and relative improvement decreasing as more peers are visited.



a)

b)

Fig. 2. Performance of runs with adapted profiles as a function of the number of peers visited in terms of a) RP@10 – where performance of the CORI baseline is given for comparison and b) relative improvement over the CORI baseline

Among the first 15 peers, the relative improvement for all profile sizes is greater than 10 %.

Finally, for comparing the results of profile adaptation not only to the CORI baseline but also to other advanced query routing strategies, Figure 3 shows the performance of the three kinds of query expansion methods described in Section 4.3.

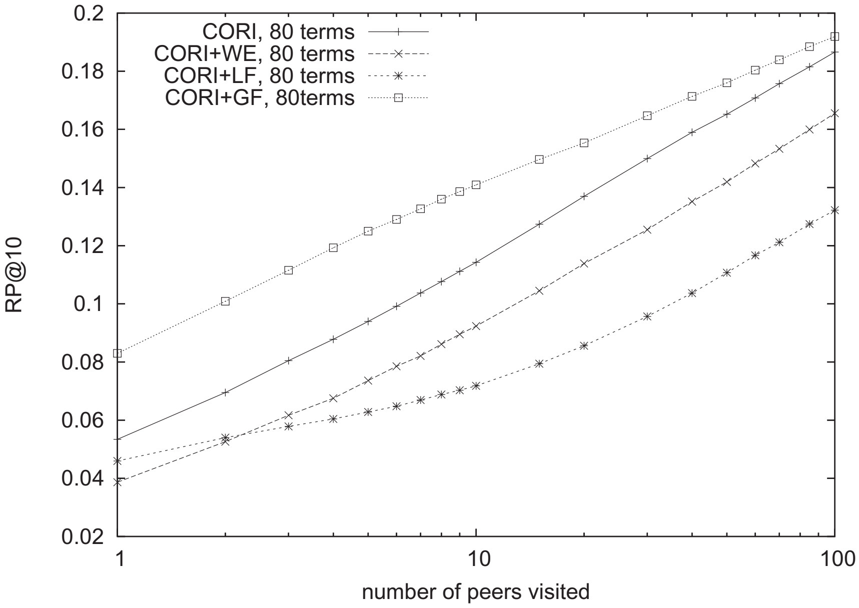


Fig. 3. RP@10 as a function of number of peers visited for web query expansion (WE), local feedback (LF), global feedback (GF) and the CORI baseline for CiteSeer with profile size 80

This clearly shows that although global pseudo feedback improves over unexpanded queries, the two expansion strategies that can be realistically applied in a distributed setting (web expansion and local feedback) are detrimental in all ranges, local pseudo feedback more so than web expansion. The situation is the same for all other profile sizes. That means that query expansion simply does not help in this context.

6 EXTENSIONS FOR PROFILE ADAPTATION

The experiments described above exploit a very simple form of user interaction, namely querying for information. The reason for this focus was the fact that corresponding usage data (i.e. query logs) are readily available for many collections – as was the case with CiteSeer. For the evaluation of approaches that rely on more complex and explicit interaction, we need data that is hard to get hold of.

However, I would like to suggest a few extensions of the previously introduced profile learning methods, even if it was not possible – for the lack of data – to evaluate these properly in this work. Here is a list of possibilities for exploiting more and more explicit user interaction:

- As it has become more or less a standard assumption that user clickthrough behaviour can be treated as a kind of noisy relevance judgments (e.g. [8]), a natural extension of the profile adaptation approach would be to derive more accurate estimates of the probability of a document's relevance by analysing if it has been clicked on or downloaded by a user. If a user downloads a document from the result list, then that is of course a stronger indication of its relevance than a simple click on it.
- Instead of estimating the probability of relevance automatically based on "implicit" user behaviour, users could be enabled to give direct and explicit feedback on the relevance of a result document, using a discrete relevance scale.
- Finally, users could be directly involved in the process of profile adaptation. Assuming a scenario where each peer is run by a single person, this approach would rely on users tagging other users. More precisely, a user who runs a peer would be informed about the neighbouring peers in the overlay (and enabled to add or remove peers from the neighbourhood), together with their profiles.

7 CONCLUSIONS

This paper has examined possibilities of learning better profiles for information sources from various kinds of user behaviour, starting from simple unigram language models derived from document contents. One particular approach, namely boosting weights of terms in profiles whenever the corresponding peer has provided useful answers to a query containing the terms, has been studied experimentally and in comparison with another state-of-the-art approach, namely query expansion. It shows high potential for making query routing more effective.

Building on these positive results, a number of extensions to the simple log-based approach have been proposed, with increasing degree of user involvement. Future work will examine these approaches in more detail, particularly focusing on their exploitation for expert search.

REFERENCES

[1] Akavipat, R.—Wu, L. S.—Menczer, F.—Maguitman, A. G.: Emerging Semantic Communities in Peer Web Search. In P2PIR '06, Proceedings of the International Workshop on Information Retrieval in Peer-to-Peer Networks, 2006, pp. 1–8.

[2] Broekstra, J.—Ehrig, M.—Haase, P.—van Harmelen, F.—Menken, M.—Mika, P.—Schnizler, B.—Siebes, R.: Bibster – A Semantics-Based Biblio-

graphic Peer-to-Peer System. In Proceedings of SemPGRID '04, 2nd Workshop on Semantics in Peer-to-Peer and Grid Computing, 2004, pp. 3–22.

[3] Callan, J. P.—Lu, Z.—Croft, W. B.: Searching Distributed Collections With Inference Networks. In Proceedings of SIGIR '95, 1995, pp. 21–28.

[4] Chernov, S.—Serdyukov, P.—Bender, M.—Michel, S.—Weikum, G.—Zimmer, C.: Database Selection and Result Merging in P2P Web Search. In Third International Workshop on Databases, Information Systems and Peer-to-Peer Computing (DBISP2P 2005).

[5] Cuenca-Acuna, F. M.—Peery, C.—Martin, R. P.—Nguyen, T. D.: PlanetP: Using Gossiping to Build Content Addressable Peer-to-Peer Information Sharing Communities. In 12th International Symposium on High Performance Distributed Computing (HPDC), 2003.

[6] Diaz, F.—Metzler, D.: Improving the Estimation of Relevance Models Using Large External Corpora. In Proceedings of SIGIR '06, pp. 154–161.

[7] French, J. C.—Powell, A. L.—Viles, C. L.—Emmitt, T.—Prey, K. J.: Evaluating Database Selection Techniques: A Testbed and Experiment. In Proceedings of SIGIR '98, pp. 121–129.

[8] Joachims, T.: Optimizing Search Engines Using Clickthrough Data. In Proceedings of KDD '02, pp. 133–142.

[9] Kalogeraki, V.—Gunopulos, D.—Zeinalipour-Yazti, D.: A Local Search Mechanism for Peer-to-Peer Networks. In Proceedings of CIKM '02, pp. 300–307.

[10] Lu, J.—Callan, J.: Content-Based Retrieval in Hybrid Peer-to-Peer Networks. In Proceedings of CIKM '03, pp. 19–206.

[11] Neumann, T.—Bender, M.—Michel, S.—Weikum, G.: A Reproducible Benchmark for P2P Retrieval. In Proceedings of First Int. Workshop on Performance and Evaluation of Data Management Systems, ExpDB, 2006.

[12] Ogilvie, P.—Callan, J.: The Effectiveness of Query Expansion for Distributed Information Retrieval. In Proceedings of CIKM '01, 2001, pp. 183–190.

[13] Shokouhi, M.—Zobel, J.: Federated Text Retrieval from Uncooperative Overlapped Collections. In Proceedings of SIGIR '07, pp. 495–502.

[14] Tsoumakos, D.—Roussopoulos, N.: Adaptive Probabilistic Search for Peer-to-Peer Networks. In Proceedings of 3rd IEEE Int. Conference on P2P Computing, 2003.

[15] Witschel, H. F.: Global Term Weights in Distributed Environments. Information Processing and Management, Vol. 44, 2008, No. 3, pp. 1049–1061.

[16] Witschel, H. F.—Holz, F.—Heinrich, G.—Teresniak, S.: An Evaluation Measure for Distributed Information Retrieval Systems. In Proceedings of ECIR '08.

[17] Witschel, H. F.: Global and Local Resources for Peer-to-Peer Text Retrieval. Dissertation, University of Leipzig, 2008.

[18] Wu, L. S.—Akavipat, R.—Menczer, F.: 6S: Distributing Crawling and Searching Across Web Peers. In Proceedings of WTAS2005.

[19] Xu, J.—Callan, J.: Effective Retrieval With Distributed Collections. In Proceedings of SIGIR '98, pp. 112–120.

[20] Xu, J.—Bruce Croft, W.: Query Expansion Using Local and Global Document Analysis. In Proceedings of SIGIR '96, pp. 4–11.

Hans Friedrich Witschel is working as a researcher at SAP Research, in the EU-funded research project MATURE (`http://mature-ip.eu`). He received his diploma and Ph. D. degrees from the University of Leipzig, his Ph. D. thesis was on peer-to-peer text retrieval. Before starting his Ph. D., he was working as a scientific assistant in a German nationally funded project on Search for text documents in large distributed systems. He is the (co-)author of numerous scientific papers at international conferences and workshops as well as of scientific book chapters and journal articles.