deskWeb2.0: Combining Desktop and Social Search

Sergej Zerr
L3S Research Center, Leibniz University of Hanover
Appelstr. 9a, Hanover 30167, Germany
zerr@L3S.de

Elena Demidova
L3S Research Center, Leibniz University of Hanover
Appelstr. 9a, Hanover 30167, Germany
demidova@L3S.de

Sergey Chernov
L3S Research Center, Leibniz University of Hanover
Appelstr. 9a, Hanover 30167, Germany
chernov@L3S.de

ABSTRACT

With availability of Web 2.0 platforms such as Flickr and YouTube, personal information is no longer locked within a user's desktop, but becomes increasingly distributed and shared across various online applications. In these settings it is important to provide a quick glance at the available personal resources and facilitate their search and selective sharing. This paper describes the challenges and requirements to be addressed in this context and presents deskWeb2.0 – an integrated environment which we currently implement towards this goal. We report the results of a small user study regarding effectiveness of such integration for different types of desktop search.

Categories and Subject Descriptors

H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval – Systems and Software

Keywords

Desktop Search, Social Search, Web 2.0.

1. INTRODUCTION

With availability of Web 2.0 platforms such as Flickr [7], YouTube [18] and Del.icio.us [5], personal space of information is no longer locked within a single desktop, but becomes increasingly distributed and selectively shared across various online applications. The Web 2.0 applications and social platforms are famous as convenient tools for sharing personal information. For example, recent report [10] shows that around 115 million bookmarks were available on the del.icio.us social bookmarking site alone in 2008. In these settings it is important to provide a quick glance not only at the user's files available locally, but also include resources shared online by the user and her friends in search results. Whereas each single Web 2.0 application is specialized in a set of predefined tasks, users would expect a single search interface over the entire set of distributed personal knowledge resources.

We illustrate this problem with a following example: Alice organizes her trip to the SIGIR 2010 conference in Geneva. She needs to retrieve relevant resources stored on her desktop, such as a sample form to authorize the trip and trip-related e-mail communication. Also, she would like to see hotel recommendations from her colleagues and multimedia resources related to places of interest in Geneva visited by her friends. Finally, she is interested in links and news shared by the other conference participants. This task would require performing search on her desktop and within each relevant social application such as Youtube, Flickr, and Del.icio.us separately. Alternatively, Alice can use *deskWeb2.0* to retrieve all these resources at once. Figure 1 presents an overview of *deskWeb2.0*.

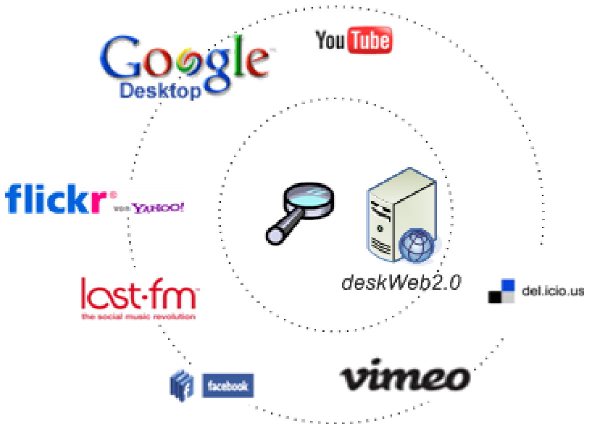

Figure 1. *deskWeb2.0* Overview

Majority of the existing desktop search applications do not support integrated search over shared social resources. A few applications such as Google Desktop Search [8] try to combine web and desktop search results, while do not support sufficient integration of search results obtained from user's accounts on social platforms. This lack of integration requires users to perform search in each social platform separately, which is a tedious task. In this paper we present *deskWeb2.0*, an integrated environment which gives its users a quick glance at the available personal resources independently of the hosting application. The contributions of this paper include: (i) a search algorithm over user's personal social network; (ii) a small-scale user study to assess how different desktop search tasks benefit from integration of social search results.

2. RELATED WORK

So far social search has not been addressed in conjunction with the desktop search problem. While modern desktop search applications allow to mix search results from the web and desktop (Google Desktop [8]) or index information on network drives (Windows Search [16], Autonomy IDOL Enterprise Desktop Search [1]), they do not search over user's Web 2.0 data, as until

SIGIR'10, Workshop on Desktop Search, July 23, 2010, Geneva, Switzerland.

now a major part of users' data was stored locally. Therefore, previous research in the area of "semantic desktop" is focused on extracting locally available metadata and storing it into a single RDF-based repository like in Haystack [13] and Gnowsis [14] systems. A recent approach implemented in the Beagle++ system [12] uses both desktop-located resources and external data proactively fetched from the Web. In contrast, our approach directly queries resources shared by the user and her friends on social services.

Social search systems allow searching for resources of different types, such as URLs, people, tags and their connections and offer ranking algorithms which take into account the structure of a social network. Hotho et al. in [9] developed ranking algorithms such as adapted PageRank and FolkRank which take network structure into account. Later, Bao et al. [2] presented alternative algorithms called SocialSimRank and SocialPageRank. Personalized ranking using social factors was considered by [3, 17]. In the current work we incorporate important ranking factors of social search to enable top-k processing.

One important task of *deskWeb2.0* is to provide an overview over the available results. To increase novelty of results and reduce the risk of user dissatisfaction, a number of schemes for diversifying results of document retrieval and database search have been proposed (e.g. [6, 15]). To give the user a quick glance of the available resources, in the current implementation we simply restrict the total number of results obtained from each service as well as the number of results retrieved from a particular user. In future work we plan to investigate alternative methods for results diversification.

3. CHALLENGES

When a large part of personal resources is distributed among heterogeneous social services, it becomes extremely difficult to provide satisfactory desktop search results. In this work we use the notion of ***social search*** to describe the search process over data gathered from user's personal networks in Web 2.0 applications, such as social bookmarking systems, blogs, forums, social network sites (SNSs), and others [3]. To integrate desktop and social search within *deskWeb2.0* we need to address several challenges discussed below.

Challenge 1: Technical and Semantic Interoperability

Both technical and semantic interoperability is required for authentication, authorization, sharing and search services of the connected platforms. Currently, such functionality is partially supported by the Web 2.0 tools using platform-dependent APIs. Given a large number of available services and possible software updates such integration becomes an essential technical problem and a laborious engineering task. Moreover, as different systems focus on specific shared data types and support different syntax, it becomes important to address these differences at a query transformation step. Finally, resources within social networks frequently change and require efficient update propagation to guarantee up-to-date search results. At the moment we address the integration problem on a purely technical level and prepare a query for each service by applying service-specific heuristics.

Challenge 2: Ranking and Aggregation of Search Results

Resources from different social platforms differ in their relevance, quality, and relation to the user significantly. Furthermore, users often share sequences of similar resources, such as photo series, such that search results can contain (near-) duplicates or similar resources in different formats. Following the ideas developed in social search [4], the relevance of resources should be influenced by the distance within the personal network, i.e. resources of closely connected peers should be ranked higher compared to resources gathered from friend-of-a-friend (FOAF). Moreover, even if all resources in a sequence have similar relevance to the query, aggregated search results should rather provide an overview over the available options. Our initial implementation of *deskWeb2.0* takes into account the distance within the personal network to facilitate top-k processing and presents a fixed number of results from each relevant platform. To this end, search result diversification, which recently attracted a lot of attention in the context of Web- and database search [6, 15] can be considered in the future work.

Challenge 3: Privacy – Preserving Resource Sharing

Web 2.0 sharing platforms represent dynamic data sources and provide up-to-date visual information about locations, people, cultural events and travelling routes. With an increasing availability of publishing applications like iPhoto [11] for the mobile devices, these resources can be almost immediately uploaded and made available within the personal network of a user on the social platforms. As such resources can be of highly private nature, they need to be handled with care. A search system is required to assist users to automatically determine the level of privacy for a particular resource. As *deskWeb2.0* searches over resources already accessible to the user and does not store them in external indices, it does not violate user's privacy.

4. SEARCH ALGORITHM

To answer a query, *deskWeb2.0* gathers user's personal resources as well as resources from the user's social network available through FOAF relationships. We model an integrated user's network as a tree, where each edge represents a friendship relationship and each node represents a user. The root node is the querying user. The children of the root node are the direct friends of the user from each connected platform. To transform a social network graph into a tree we apply a greedy algorithm which selects the shortest paths within the graph. To retrieve up-to-date diverse search results, we implement a query propagation algorithm presented in Algorithm 1. Algorithm 1 traverses the tree in a breadth first manner. As a node can possibly contain either too many or too few results, the goal of Algorithm 1 is to obtain a balanced result set giving the user an overview over the available results. To decide on the number of results to be retrieved from a node *n*, it weights *k* in top-k with the relevance of the node *n*. In case the node does not contain enough results for a query, it propagates the remaining number of results to the *n*'s children.

```
getTopk(keywords, root, k, min_relevance, results){
 priorityQueue<relevance> queue;
//compute the max number of results to be returned from the node
 root.maxresults=root.relevance * k;
 queue.enqueue (root);
 while (true)  {
        node = queue.dequeue();
        //check relevance threshold
         if (node.relevance < min_relevance) break;
         node.results=node.query(keywords, node.maxresults);
        //collect results from the node
        results.add(node.results);
        if (results.size()>=k) break;
        for (friend: node.friends){
        //propagate remaining results to child's nodes
        friend.maxresults=node.maxresults-node.countresults;
        queue.enqueue(friend); }} }
```

Algorithm 1. *deskWeb2.0* Search

The relevance of a resource in the integrated social network of *deskWeb2.0* depends not only on the content of the resource, but also on the global importance of its owner within the network and the strength of the relationship between the owner and the querying user. This relevance can be computed using Equation 1:

$$rel(r,n,q) = rel(n, n.network)\cdot rel(root->n)\cdot rel(r.content, q), \qquad (1)$$

where relevance of the resource *r* from node *n* with respect to query *q* is the relevance of the node *n* in the context of its social network *n.network*, rel(*root*->*n*) is the relevance of the node *n* with respect to the querying user *root* and rel(*r.content*, *q*) is the relevance of the content of *r* with respect to the query *q*. According to Equation 1, each node within user's network can be weighted independently of the query which supports efficient top-k query processing.

5. EVALUATION

To find out how users search personal resources using currently available tools we carried out a small questionnaire. We asked 22 graduate students from Computer Sciences department to tell us which tool they use to find resources on their desktop. Majority of the Windows users (10 out of 12) use native Windows Desktop Search tool; 3 participants uses other tools, such as Spotlight or "find" command for Linux; 9 users do not use desktop search tools. As our current implementation relies on Google Desktop Search to retrieve local search results, we had to limit our user study to few people who installed it.

In our evaluation of *deskWeb2.0* we focused on two research questions: **Question 1**: Does search over social services contributes to desktop search with respect to the relevance of results? **Question 2**: Which types of desktop search such as location, people and general information finding benefit from social search and which do not?

To answer these questions we performed a small user study. Our participants were five students from Computer Sciences department, who had Google Desktop Search tool installed. We selected four tasks for the users to perform, each including three search types. Each task required the user to retrieve certain information from the integrated environment of *deskWeb2.0*:

T1. *Collect information for a business trip*; **T2**. *Prepare a tutorial on a topic of interest*; **T3**. *Organize a short-distance weekend trip with friends/family*; and **T4**. *Organize a party*. Each task included the following search types: **A**. *Find contact details of a person*; **B**. *Find location information*; and **C**. *Find general information*. For each task and search type, we asked users to issue a keyword query of their choice. For every query, we presented the user with two lists of results, one containing top-5 results from Google Desktop Search, and another list containing up to five results from each Web 2.0 service on which the user had an account. We asked users to rate the results on a 3-point scale as "relevant", "less relevant", or "non-relevant". The users had one or two active accounts on social services supported by our prototype so quantities of desktop and social results were comparable.

To answer **Question 1**, we computed the macro-averaged Normalized Discounted Cumulative Gain (nDCG) in three result lists: desktop, Web 2.0, and a merged list. To compute NDCG we ranked each list by TF-IDF scores. On Figure 2 we present the nDCG values for top-5 results averaged over the participants. As we can see from Figure 2, although absolute nDCG values of Web 2.0 results is lower than the values obtained by the desktop search, combination of desktop and social search results increases the gain of desktop search for all k>2 by about 6% on average. We also report the results per each task T1 – T4 to see if there are any specific situations in which social search is useful. For readability reasons we split these results into two plots i.e. Figure 3 and Figure 4 and present only desktop and merged results. From the task-wise presentation we observe that tasks T1 and T3, both related to travelling preparations, only modestly benefit from merging with social search results. In contrast, task T2 about tutorial preparation shows stable and significant improvement over pure desktop search. Finally, party-planning task T4 shows about double nDCG improvement over regular desktop search. Therefore, we conclude that **Question 1** could be answered positively and search over social services significantly complements relevance of the desktop search.

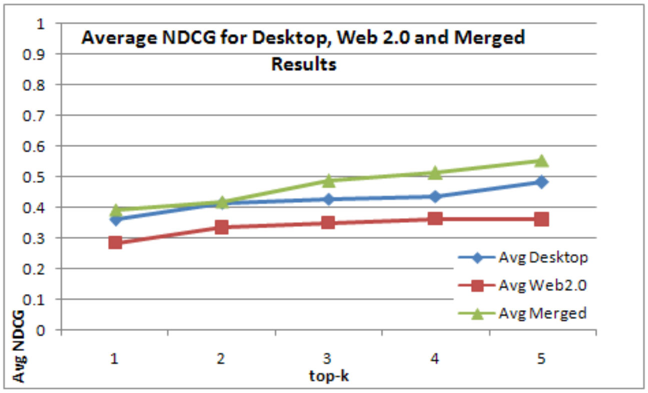

Figure 2. Average nDCG of all Tasks

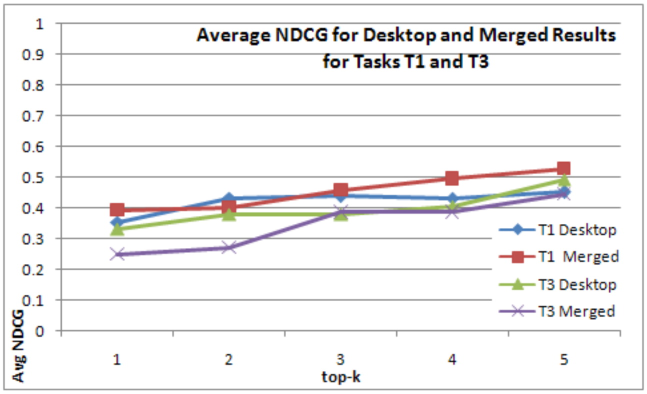

Figure 3. Average nDCG for Tasks T1 and T3

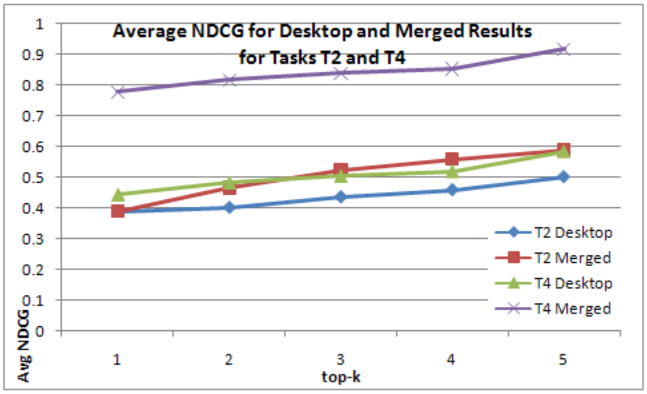

Figure 4. Average nDCG for Tasks T2 and T4

To answer **Question 2**, we considered nDCG in desktop and merged results for each type of search such as people (A), location (B) and general information (C) separately. Figure 5 presents nDCG results averaged over the users and search types. We observe that both people and general information search profit from the mixture of desktop and social search. nDCG for people search improved by about 17% and general information finding by 10%. On the contrary, nDCG of location search in the merged list decreased. Since nDCG of the social search results for location search was much lower than that of the desktop search their mixture did not provide any extra advantage. This result also explains the modest improvements in travelling-related tasks T1 and T3 where location search is important.

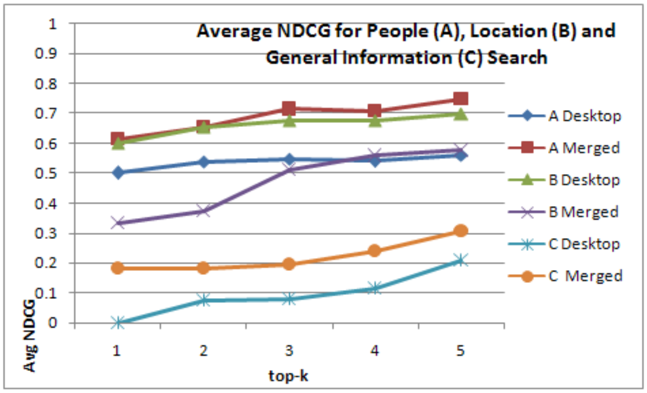

Figure 5. Average nDCG for three Types of Desktop Search

Our answer for the **Question 2** is that general information search benefits the most from social search, possibly, due to the low desktop search effectiveness. Desktop search for people finding is already very effective, but it is also significantly improved by the Web 2.0 results. Location search degrades when using social search results, we assume that users did not have relevant information for this search type in the current networks.

6. CONCLUSION

Nowadays, personal resources are no longer stored within a single desktop, but are increasingly shared across social platforms. This work presents the first steps towards integrating desktop search with social search over shared personal resources. We identified some challenges to be addressed and developed a sample *deskWeb2.0* application. A small user study demonstrated that search over social resources increases overall search accuracy. It also suggests that people finding and search for general information benefit from such integration, while search for locations is more effective using desktop search alone. In the future we plan to perform a larger user study. Also, we would like to investigate diversification of search results in this context.

7. REFERENCES

[1] Autonomy IDOL Enterprise Desktop Search http://www. autonomy.com/content/Products/enterprise-search/index.en.html

[2] S. Bao, G. Xue, X. Wu, Y. Yu, B. Fei, and Z. Su. Optimizing web search using social annotations. In WWW 2007.

[3] M. Bender, T. Crecelius, M. Kacimi, S. Michel, T. Neumann, J. X. Parreira, R. Schenkel, and G. Weikum. Exploiting social relations for query expansion and result ranking. ICDE Workshops, 2008.

[4] D. Carmel, N. Zwerdling, I. Guy, S. Ofek-Koifman, N. Har'El, I. Ronen, E. Uziel, S. Yogev, S. Chernov. Personalized social search based on the user's social network. In Proc. of CIKM 2009.

[5] Delicious: http://delicious.com/

[6] E. Demidova, P. Fankhauser, X. Zhou, W. Nejdl. DivQ: Diversification for Keyword Search over Structured Databases. In Proceedings of SIGIR 2010.

[7] Flickr: http://www.flickr.com/

[8] Google Desktop Search: http://desktop.google.com/

[9] A. Hotho, R. Jäschke, C. Schmitz, and G. Stumme. Information retrieval in folksonomies: Search and ranking. In Y. Sure and J. Domingue, editors, The Semantic Web: Research and Applications, volume 4011 of LNAI, Heidelberg, June 2006, Springer.

[10] P. Heymann, G. Koutrika, and H. Garcia-Molina. Can social bookmarking improve web search? In Proc. of WSDM. ACM, 2008.

[11] iPhoto: http://www.apple.com/ilife/iphoto/

[12] E. Minack, R. Paiu, S. Costache, G. Demartini, J. Gaugaz, E. Ioannou, P.-A. Chirita, and W. Nejdl. Leveraging personal metadata for desktop search – the Beagle++ system. In Journal of Web Semantics, 2010.

[13] D. Quan, D. Huynh, D. R. Karger, Haystack: A Platform for Authoring End User Semantic Web Applications. ISWC'03.

[14] L. Sauermann, Using Semantic Web Technologies to build a Semantic Desktop,Master's thesis, TU Vienna (2003).

[15] J. Wang, J. Zhu. Portfolio Theory of Information Retrieval. In Proceedings of the SIGIR 2009.

[16] Windows Desktop Search: http://www.microsoft.com/windows/ products/winfamily/ desktopsearch/default.mspx

[17] S. Xu, S. Bao, B. Fei, Z. Su, and Y. Yu. Exploring folksonomy for personalized search. In Proceedings of SIGIR, ACM, 2008.

[18] Youtube: http://www.youtube.com/