Transparent and User-Controllable Personalization For Information Exploration

Peter Brusilovsky
School of Information Sciences, University of Pittsburgh
135 N. Bellefield Ave.
Pittsburgh, PA 15256, USA
peterb@pitt.edu

ABSTRACT

Personalized Web search has emerged as one of the hottest topics for both the Web industry and academic researchers. However, the majority of studies on personalized search focused on a rather simple type of search, which leaves an important research topic – the personalization in exploratory searches – as an under-studied area. In this paper, we present a study of personalization in task-based information exploration using a system called TaskSieve. TaskSieve is a Web search system that utilizes a relevance feedback based profile, called a "task model", for personalization. Its innovations include flexible and user controlled integration of queries and task models and on-screen visualization of task models. Through an empirical study using human subjects conducting task-based exploration searches, we demonstrate that TaskSieve pushes significantly more relevant documents to the top of search result lists as compared to a traditional search system. TaskSieve helps users select significantly more accurate information for their tasks, allows the users to do so with higher productivity, and is viewed more favorably by subjects under several usability related characteristics.

1. INTRODUCTION

It is commonly accepted that lookup search is just one of several types of searches performed by Web users. Marchionini [6] calls searches "beyond lookup" as *exploratory searches*, which can be further distinguished as *search to learn* and *search to investigate*. Exploratory search assumes that the user has some broader *information need* that cannot be simply solved by a "relevant" Web page, but requires multiple searches interleaved with browsing and analyzing the retrieved information. The research on supporting exploratory search attracts more and more attention every year for two reasons. On one hand, the number of users engaged in exploratory search activities is growing. With the growth of information available on the Web, almost any Web user performs searches "beyond lookup" on such occasions as planning a vacation or choosing the most relevant product (i.e., digital camera). Moreover, some classes of users, such as intelligence analysts, perform multiple exploratory searches every day as a part of their job. On the other hand, traditional search systems and engines working in a mode "query – list of results" provide very poor support for exploratory search tasks [6]. Neither is it easy for users to formulate a query when it is not really clear what they are looking for, nor is the result presentation in the form of a linear list helpful to make sense of the retrieved information.

Our team investigated the issue of exploratory search in the context of DARPA's GALE project. Our goal was to develop a better information distillation interface for intelligence analysis. We focused on personalized search expecting that adaptation to an analyst's global task beyond a single query may help our system to bring better results to the analyst's attention.

Personalized search emerged as one of the hottest topics for both the Web industry and academic researchers [7]. Unlike traditional "one-size-fits-all" search engines, personalized search systems attempt to take into account interests, goals, and preferences of individual users in order to improve the relevance of search results and the overall retrieval experience. In the context of a tight competition between search engines and technologies, personalization is frequently considered as one of the technologies that can deliver a competitive advantage.

We expected that personalized search will be appreciated by users engaged in information explorations and will allow them to achive a sizeable performance increase. However, an evaluation of our personalized intelligence analysts discovered that traditional personalized search does not provide the proper level of support in an information exploration context. While appreciating the value of personalization, the analysts repeatedly asked for an interface that provides "more transparency" and "more control" over the search process. Unfortunately, neither transparency, no control are provided by the traditional personalized search systems provide. Personalization works as a black box, which starts a query produces a user-adapted list of results with no direct user involvement. Inside this black box, the personalization engine applies a user profile either to generate query expansion or to reorder search results [7].

In our recent work we explored an alternative approach to implementing personalized search, specifically geared to information exploration context. In our TaskSieve system [2], we attempted to implement personalization as an information exploration tool, which offers the user both: a reasonable control over the process and a better transparency of its mechanism. We consider transparency as an important component of user control: without clear understanding of the process, which is supported by transparency, an effective control is hardly possible. The remaining part of the paper briefly presents the components of TaskSieve interface, which demonstrates our vision of transparent and controllable personalization. The results of TaskSieve evaluation can be found elsewhere [2].

2. TaskSieve: A PLATFORM FOR TASK-BASED INFORMATION EXPLORATION

2.1 A Transparent Task Model

Unlike the majority of known personalized search systems, TaskSieve aims to support the task-based exploratory search process. In place of a traditional model of user interests,

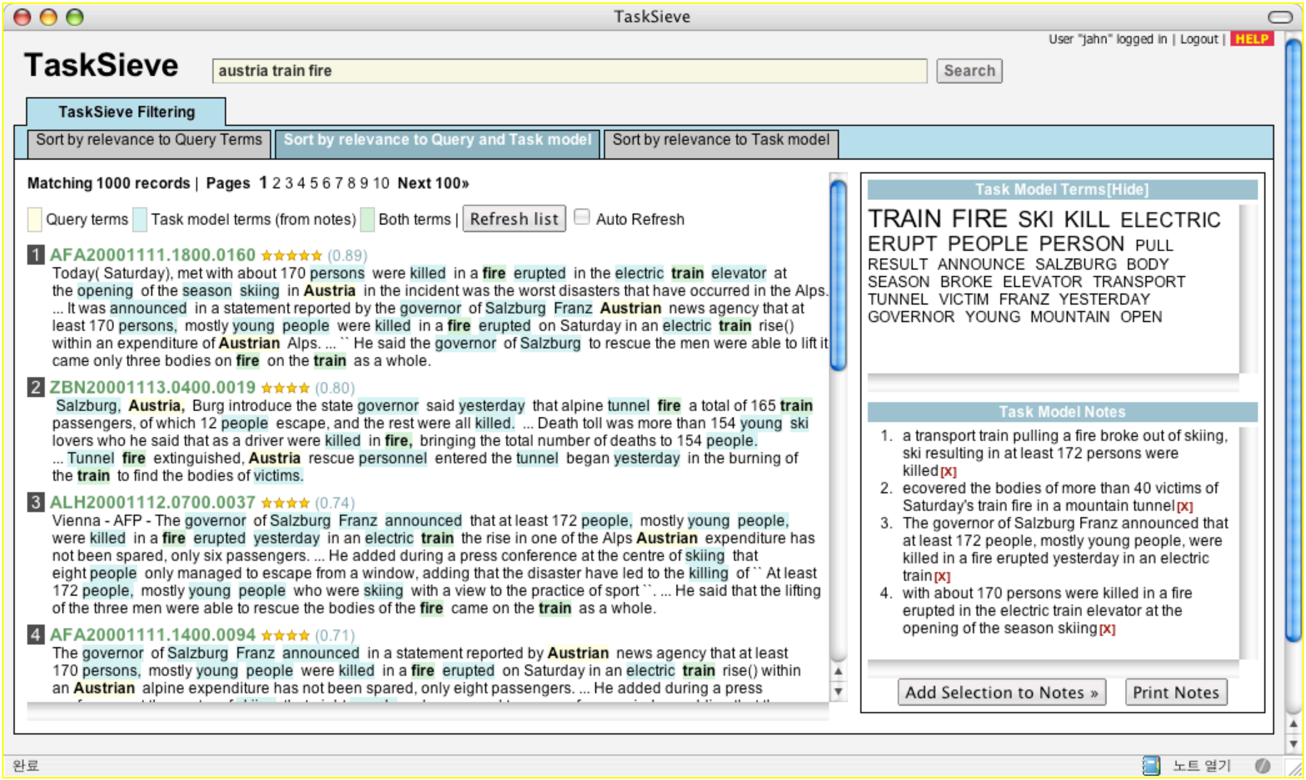

Figure 1 TaskSieve interface

TaskSieve applies a more focused *task model*, which attempts to accumulate information about the task explored by the user. A task model is a relatively short-term model in comparison with a long-term model of user interests, yet it can support the user over a lengthy sequence of queries (frequently spread over several sessions) as long as the user is focused on a specific task. The model is constructed unobtrusively while the users are interacting with the system. There is no task description to enter, as in AntWorld [5] or SERF [4]. The user simply starts working on a new task by entering the first query and processing the list of initial, but not yet adapted, search results. Standard stemming and stopword removal procedures are applied to these task model vectors. Among the hundreds of terms from the user notes, the top 300 important terms are selected according to their TF-IDF weights in the document corpus.

TaskSieve was designed to assist users who perform exploratory searches reasonably often, i.e., it focuses on relatively experienced searchers up to the level of professional information analysts. These users appreciate more powerful and sophisticated information access tools; but as we learned from our earlier work on adaptive filtering [1], they also want to be in control of the system's work and highly value the transparency of the system mechanisms. This requirement contradicts the traditional approach taken by personalized search systems, which tend to make personalization decisions without user consent and hide the underlying personalization mechanism. Unlike these systems, TaskSieve attempts to make the personalization transparent. It starts with using a relatively simple, but easy to understand task model form: weighted term vectors. In addition, it makes the task model visible to the user through the model viewer (upper right in Figure 1). The viewer shows terms, which form the task model, sorted by their importance (weight). A larger font size is used for more important terms. The model visualization is kept up-to-date according to the task model changes. This visible task model is expected to help users to understand the task-based engine of TaskSieve; however, users who consider the model less useful or need more space for other parts of the interface can hide the viewer at any time.

2.2 Controllable Personalized Ranking

As in many other personalized search systems, TaskSieve uses the post-filtering approach to personalized search results, using the task model to re-rank the plain search results retrieved by a search engine (**Error! Reference source not found.**). The idea of re-ranking is to promote documents, which are more relevant to the user task as measured by their similarity to the task model. For transparency reasons, TaskSieve uses the traditional linear approach to combine query relevance and task relevance:

(1) Retrieve documents along with their relevance scores by submitting the user query to a search engine.

(2) Calculate similarity scores between retrieved documents and the model.

(3) Calculate combined score of each document by equation (1).

$$alpha * Task\_Model\_Score + (1 - alpha) * Search\_Score \quad (1)$$

(4) Re-rank the initial list by the combined score from step 3.

TaskSieve uses Indri[1] as a search engine and normalizes its scores, dividing by the maximum score (score of the rank 1 item) of the

[1] http://www.lemurproject.org/indri

corresponding list (step 1). Task model scores are calculated by measuring the similarity between each document vector and the task model vector. We use BM25 **Error! Reference source not found.** for this task (step 2) and the scores are also normalized.

In equation (1), *alpha* controls the power of the task model. It can vary freely from 0 to 1. The traditional approach is to fix *alpha* either ad-hoc, or by learning the "optimal" value and using this value to fuse all search results. We believe this approach contradicts the desire of our target users to be "in control", and instead give the control over the fusion to users. TaskSieve allows the users to alternate among three preset ranking options: "Sort by relevance to Query Terms", "Sort by relevance to Query and Task model", and "Sort by Relevance to Task Model" (which correspond to *alpha* values 0, 0.5, and 1.0 respectively). If *alpha* is 0, the ranking is the same as plain search. If *alpha* is 1.0, then the search rank is completely ignored. If *alpha* is 0.5, which is the default, the system considers equally the importance of query and task.

Figure 1 shows an example of the task-based ranked list (lower left in the screen). A user enters a query "austria train fire". Important task model terms such as "TRAIN", "FIRE", "SKI", and "KILL" were extracted from the user notes in order to re-rank the original search result to the query "austria train fire" generated from the baseline search engine. Just above the ranked list, there are three tabs labeled with three ranking options explained above. Users can explore different query terms and control the task-based post-filtering engine in order to complete their tasks.

2.3 Using Notebook for Task Model Update

In addition to the innovative ways of using the task model, TaskSieve explores a new transparent approach to updating this model. This approach is based on the idea of a *notebook*. A notebook is a collection of document fragments (which we call *notes*) extracted and saved by the user. From one side, the notebook supports the user's need to collect the most important information for further processing. A note collection tool is frequently used in the process of information exploration (analysts call it a "shoebox"). From the other side, the content of the collected notes represents the task much better than the documents from which they are extracted. It allows TaskSieve to use the content of the saved notes to increase the quality of modeling in comparison with existing personalized search systems.

TaskSieve encourages the user to take notes and make this process very simple. The users can highlight any text from the search snippets or whole document and add it to the notebook by a single button click. When a new note is saved, it is displayed in the notebook (lower right in Figure 1). Each note can be removed by clicking on the "X" beside it if the user doesn't think she needs it anymore.

Every action in the notebook (adding and removing) instantly affects the task model – the weights of the task model terms found in the added or removed note are increased or decreased correspondingly. The important task model terms in the task model viewer are immediately updated to reflect the new set of weights. The ranking of the current search result list can also be updated immediately after each task model change if *Auto Refresh* is checked. However, this option is switched off by default, because our previous studies in a similar context of information filtering demonstrated that automatic update of ranking confuses users and causes performance decreases [3]. Therefore, TaskSieve offers a "Refresh list" button, allowing the user to re-rank the search results according to the current state of the task model whenever it is most convenient for her.

3. STUDY RESULTS AND FUTURE WORK

We conducted an empirical study with human subjects using TaskSieve for task-based exploration searches. The study demonstrates that TaskSieve – compared to a traditional search system – can utilize the information available in the task model to return significantly more relevant documents at the top of the ranked lists. The data also show that the average precision values of the baseline system's ranked lists at the last 10 minutes is still lower than that of the experimental system's first 10 minutes. This shows that the improvement obtained through task model is even higher than that through human users learning about the search topic and the retrieval system over the time.

The study also shows that TaskSieve can help user performance, too. TaskSieve's users were not only able to select notes that contained significantly more relevant information, they also can select more notes even during the first 10 minutes of the search session when they were still relatively unfamiliar with the search tasks. This demonstrates that TaskSieve significantly improved the productivity of the users' searches.

The flexibility in controlling the integration mode between queries and the task model also demonstrates its usefulness. First, we observed subjects switching among the different modes in their searches. Second, the searches with the half-half mode produced the best results. Third, the searches in query-only mode produced better results than the baseline, which indicates that the users really mastered the preset manipulations and used the appropriate mode for different searches. Finally, it is clear that none of the modes significantly dominates all the searches. All of these indicate that it really makes sense for TaskSieve to let users decide the best mode for their searches.

4. ACKNOWLEDGMENTS

This paper is partially supported by DARPA GALE project and by the National Science Foundation under Grant No. 0447083.

5. REFERENCES

[1] AHN, J.-W., BRUSILOVSKY, P., GRADY, J., HE, D., SYN, S.Y. Open user profiles for adaptive news systems: help or harm? in Proceedings of the 16th international conference on World Wide Web, WWW '07 (Banff, Canada, May 8-12, 2007), ACM 11-20.

[2] AHN, J.-W., BRUSILOVSKY, P., HE, D., GRADY, J., LI, Q. Personalized Web Exploration with Task Models in Proceedings of the 17th international conference on World Wide Web, WWW '08 (Beijing, China, April 21-25, 2008), ACM 1-10.

[3] HE, D., BRUSILOVSKY, P., GRADY, J., LI, Q., AHN, J.-W. How Up-to-date should it be? The Value of Instant Profiling and Adaptation in Information Filtering in Proceedings of the 2007 international conference on Web Intelligence, WI '07 (Silicon Valey, CA, USA, November 2-5, 2007), IEEE 699-705.

[4] JUNG, S., HARRIS, K., WEBSTER, J., HERLOCKER, J.L. SERF: Iintegrating human recommendations with search in Proceedings of ACM 13th Conference on Information and Knowledge Management, CIKM 2004 (Washington, DC, November 8-13, 2004 571-580.

[5] KANTOR, P.B., BOROS, E., MELAMED, B., MEÑKOV, V., SHAPIRA, B., NEU, D.J. Capturing human intelligence in the net, Communications of the ACM (2000) 43, 8, 112-116.

[6] MARCHIONINI, G. Exploratory search: From finding to understanding, Communications of the ACM (2006) 49, 4, 41-46.

[7] MICARELLI, A., GASPARETTI, F., SCIARRONE, F., GAUCH, S. Personalized search on the World Wide Web, In: Brusilovsky, P., Kobsa, A., and Neidl, W., Eds. The Adaptive Web: Methods and Strategies of Web Personalization. Springer-Verlag, Berlin Heidelberg New York, 2007, 195-230.