Focus on Results: Personal and Group Information Seeking Over Time

Gary Marchionini, Robert Capra, and Chirag Shah
School of Information and Library Science
University of North Carolina at Chapel Hill
100 Manning Hall
march@ils.unc.edu, rcapra3@unc.edu, chirag@unc.edu

1.INTRODUCTION

Information seeking is a fundamental human activity that is applied to an enormous range of information needs and exhibits diverse sets of individual behavioral nuances. Information needs range from fact retrieval to life-long interests in complex constructs and information-seeking behaviors range from brute force exhaustive search to sophisticated heuristics (e.g., building block, successive fraction, pearl growing, e.g., Hawkins & Wagers, 1982) and stochastic estimations. Today's search engines leverage content, links, metadata, and context such as time and place to return information based on searcher queries or selections. It is left to the information seeker to examine, interpret, and manage results independent of the search system, a condition that we aim to address here.

2.THE PROBLEM OF RESULTS

It is well known that people spend much more time examining results (both result sets and specific documents/pages) than composing queries (e.g., see Weinreich et al., 2007), however, the main emphasis of search engines is query processing, leaving the results examination to information seekers. Some search systems provide some results support. For example, Clusty (clusty.com) organizes results in clusters and the Cuil (cuil.com) provides spatial layouts of top-ranked search results. Coyle & Smyth (2007), Shneiderman and his colleagues (1994) and others have emphasized design of systems that support the entire search process and over the years, we have aimed to couple queries and results through highly interactive interfaces (e.g., the Relation Browser; Marchionini & Brunk, 2003; Capra & Marchionini, 2007). In this paper we focus on a framework for results management that will support searches over multiple sessions and possibly in collaboration.

Whereas most user-centered IR research focuses on query formulation and reformulation, we propose making the results of search the focal point of our work. By taking this novel approach to exploratory search, we aim to fill a gap between query-oriented IR and the personal/group information management systems (PIM and GIM, e.g., Erickson, 2006) that support information use. We propose a result space support system as a way to attack the multi-session exploratory and collaborative search problems and fill this gap in current research and development. To this end, we describe a result space architecture and outline one possible prototype based on this architecture that supports managing and optionally sharing result sets and items. Objects in the space include attributes such as search genesis (e.g., query), related objects (explicitly tagged, automatically linked), and temporal status (e.g., changes over time) and can be sharable individually or in aggregate.

Information seeking often takes place over multiple sessions. Current practices to deal with this include ad-hoc strategies. Email to self (Jones, et al. 2001; Whittaker, et al. 2006) has been documented as a particularly common strategy due in part to its ease of re-access from any location. Other strategies for re-access include bookmarks, saving and printing documents, and relying on being able to relocate information using search engine (Jones et al, 2001; Bruce et al. 2004; Aula et al. 2005). Studies have found that users struggle to make these ad-hoc strategies work for their needs and that personal information management is a challenge for users (Aula et al. 2005; Jones et al. 2001; Bruce et al., 2004). We aim to create a framework and tools for analyzing, saving, managing, and re-using results that will help overcome these ad-hoc strategies.

In the early days of online searching, professional intermediaries adopted techniques to reuse searches as they served many researchers with common interest (e.g., the Dialog search system allowed intermediaries to save sessions and query strategies more than 30 years ago). Komlodi's dissertation revealed the complexities of search history support in her study of searchers in law firms (Komlodi, 2002). She used participatory design to create prototype user interfaces that were in turn evaluated by legal searchers (Komlodi et al., 2007). She defined a search history framework with six primary components each with a hierarchical collection of factors: (scope of search history [21 factors at 3 levels], search context [28 factors at 4 levels], search history data [140 factors at 8 levels], search result management [24 factors at 4 levels], search history use [78 factors at 6 levels], and design features [80 factors at 5 levels].

Once relevant information is found, there are a variety of tools and services (e,g., RefWork, Zotero, Google Notebook, and Firefox Scrapbook) that support collecting and reorganizing search results. On-line tagging, bookmarking and social networking sites such as del.icio.us provide users with basic tools for storing bookmarks, tagging them, and sharing them with others. However, these systems treat the results as discrete and static objects and disassociate them from the queries that generated them. We aim to close this gap and closely couple queries, result sets, and results (items) to facilitate reuse and ongoing information seeking over time.

Highly interactive search system styles such as dynamic queries (Alberg et al., 1992; Shneiderman, 1994) bridge the gap between discrete queries and result pairs to tightly couple queries and results and thus shift the focus from single-query retrieval to session-oriented retrieval. One of our primary goals in the work proposed here is to support tight coupling of inter-session searches. We moved in this direction with our Govstat project (find what you need, understand what you find, Marchionini et al.,2003) and aim to press further on this trajectory to more tightly couple queries and results over multiple sessions.

3.THE RESULTS FRAMEWORK

A general set of desiderata and vision for the system components follows. Such systems should allow people to easily:

1. Add results from new searches and result sets to their results space (perhaps coming from different sources and via automated processes);

2. Add annotations and tags to results, result sets, and queries;

3. Monitor changes to results, result sets, and queries over time;

4. Dynamically manipulate multiple result sets and queries to investigate overlaps, disjunctions, and changes over time; and

5. Selectively reuse and share results, result sets, and queries and their tags and annotations.

Figure 1 presents a schematic view of the Result Space, which consists of three dimensions: results, sessions, and users. A given cell in this cube consists of a set of Result Frames (RFs) for a result object (e.g., a web page, PDF file, video file) for a single session by a single user. Note that for queries that do not yield saved results, if the users wants nonetheless to save the query for future reuse, we will index a "no saved result" (null) entry that includes the query and other generative and contextual information. This will allow the user a more continuous history option within a single data model.

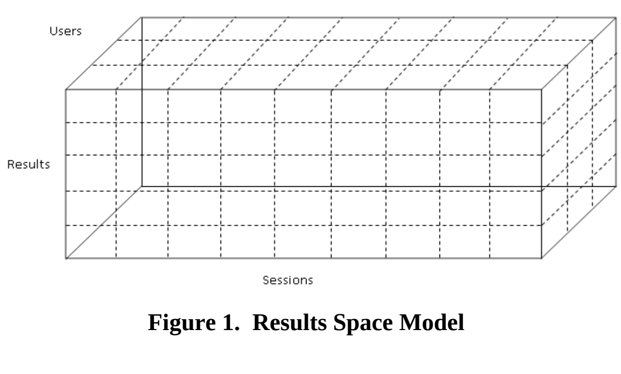

Figure 1. Results Space Model

A vertical stack of RFs represent a set of different results from a single session by a single user. An entire vertical slice represents the RFs for multiple users in a collaborative session (note that the session might be synchronous or not in practice, however, the figure suggests contemporaneous sessions). A horizontal (left to right in figure 1) stack of RFs represents a result that occurs in multiple sessions (e.g., revisiting or refinding, reusing the result in another session). The RFs are 'sets' because we want to explicitly store changes to a result that occur over time either by the user within a session (e.g., finding and saving the same result with different queries in a session) or more importantly, without user intermediation after a session (e.g., tagging or reuse by a collaborator, item revisions, or rank changes in the result set for the generating query). Thus, we treat results as dynamic objects within the result space that are active even when the user is not attentive to them and show these changes when the user is attentive. Considering RFs across the sessions and across the users supports the system's history mechanism and group information processing mechanism respectively.

A RF consists of a specific result (e.g., a web page, PDF file, video file) with various attributes and subspaces associated with it. Each RF has attributes: rank, tags, and notes. Rank stores the rank of the result in the query that generated it. Tags are provided by the user and/or other collaborative users in the group and are similar to the tags used in social network and bookmark system. Notes offer users a way to annotate results with personally meaningful contextual information. Unlike tags, notes allow free-form text that can include descriptions of the relevance of the result to the person or project, information for collaborative group members, a plan for using the result, or other annotations.

In addition to the attributes described above, each RF also contains the following subspaces: query, related, facets, world, and social. These subspaces can have attributes of their own. The query subspace contains attributes of the query that generated the result. These attributes are the query string, the source where the query was executed (e.g. Google, ACM Digital Library), and a listing of the top 100 results for the query (this number will be adjustable) from that source. The related subspace could contain items recommended by the source as being related to the result object such as explicit recommendations or outlinks.

Many result objects will have associated faceted metadata that can be stored in a facet subspace. For instance, video objects might have metadata facets for duration, genre, creation date, and source. Facets consist of pairs of a facet name (e.g. "size") and a counterpart value (e.g. "Medium"). Faceted metadata may come from the data source itself, may be obtained from shared collaborative information, or may be automatically generated by classification engines. Our research group has extensive experience building interfaces and classification engines to support faceted search (Capra et al., 2007; Zhang and Marchionini, 2005; Efron, et al., 2004; Marchionini and Brunk, 2003).

A world subspace represents external attributes that are obtained from sources other than the source where the information was located. In most cases, the other sources will be Web-based services. For instance, if the result is a blog entry, we may consult a web search engine to obtain information about the PageRank of the blog page and the inlinks to that blog.

Collaborative contributions about the result are captured in the social subspace. The community around the user may be a close work group or a broader social network. For instance, if the result is a book, the social subspace might record how other people have rated the book and include all the associated reviews from a social network. A permissions mechanism will be implemented at the RF, subspace, and attribute levels so that users can selectively share at fine grains. Each of these attributes and subspaces of results can be used to organize, filter, manage, and re-use result spaces.

4.PROTOTYPE IDEAS

In order to support these user activities, we outline ideas for a Results Space (RS) system. Such a system must manage shared access at different granularities (allow a user to specify which items and attributes are shared with whom) and automatically update contextual information over time. Information will be gathered from client devices (upon user initiation) as the user engages in web browsing and searching. To support the ability to start an information seeking session on one device and continue it on another device or in later session, information gathered during the search will be stored on a server. We intend to build the client and server software on top of established open-source software components and provide simple installers so that users could easily use the RS system on a local server or intranet.

Storing queries, result sets, and annotations on a central server has a number of privacy issues for many users and organizations. Emerging bookmark and web clipping notebook services (e.g. del.icio.us and Google notebook) require users to set up accounts and store information on their central servers. An alternative approach taken in the UCAIR project (Shen et al., 2005a, 2005b) is to store search history on the client machine, however, this does not allow people to use multiple platforms for their ongoing work. There is a classic tradeoff between supporting remote access (storing information on a server) and providing more privacy and security (store information in only one place – on the user's PC). One compromise that many companies use is to host their own servers (e.g. corporate email servers) to provide remote access while gaining a level of control of the privacy and security for their organization. In fact, this type of intra-net level use is one of the situations where we envision the collaborative aspects of the RS system being most useful –knowledge workers on a project team within an organization collaboratively conducting searches and synthesizing results in order to create their work products. The RS system outlined here must provide the tools and central coordination needed to support such collaborative research work.

One of the interfaces we anticipate including in the RS system is a web browser toolbar that allows users to interact with current search results as well as with previous result sets. Toolbars are a commonly used, unobtrusive interface that can provide a lightweight way to add and annotate results while also providing controls that can expand or use the main web browsing space to support additional interactions such as visualizations.

In the RS system, the individual result items found by the user as part of their information seeking will be the main focal objects. However, individual results, result sets, and queries will all be first-class objects in the proposed architecture. This means that they can all be stored, manipulated, composed, and inspected as part of the system. This style of architecture will allow different controls, visualizations, and operations to be easily developed and "plugged-in" to the RS system. We have experience developing complex query and result set models using this style of architecture from our prior work on the Relation Browser (Capra and Marchionini, 2007a, 2007b) and Context Miner (Shah & Marchionini, 2007) systems. In the work proposed here, we will extend the architecture and interfaces to support: 1) multiple sessions (extending the model across time), 2) multiple devices (extending the model to support full and limited feature sets), 3) annotations and connections to results, result sets, and queries, and 4) collaborative views and reuse of the data.

5.ACKNOWLEDGMENTS

This work was supported by NSF grant IIS 0812363.

6.REFERENCES

[1] Ahlberg, C., Williamson, C. and Shneiderman, B. (1992). Dynamic queries for information exploration: An implementation and evaluation. In Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems, pp. 619-626.

[2] Aula, A., Jhaveri, N., Käki, M. (2005). Information Search and Re-access Strategies of Experienced Web Users, In Proc. of WWW 2005, Chiba Japan, 583-592.

[3] Bruce, H., Jones, W., and Dumais, S. (2004). Keeping and re-finding information on the web: What do people do and what do they need? In Proc. ASIST 2004, Chicago, IL, Information Today, Inc., October, 2004.

[4] Capra, R. and Marchionini, M. (2007). Faceted Browsing, Dynamic Interfaces, and Exploratory Search: Experiences and Challenges. Proceedings of the Workshop on Human-Computer Interaction and Information Retrieval (Cambridge, MA, October 23, 2007), 7-9.

[5] Capra, R., Marchionini, G., Oh, J. S., Stutzman, F., and Zhang, Y. (2007). Effects of structure and interaction style on distinct search tasks. Proceedings of the 2007 Conference on Digital Libraries (Vancouver, BC, Canada, June 18 - 23, 2007).

[6] Coyle, M. & Smyth, B. (2007). Supporting intelligent web search. ACM Transactions on Internet Technology. 7(4), 1-31.

[7] Efron, M., Elsas, J., Marchionini, G., and Zhang J. (2004). Machine learning for information architecture in a large governmental website. Proceedings of the 4th ACM/IEEE Joint Conference on Digital Libraries (Tuscon, AZ, June 7-11, 2004), 151-159.

[8] Erickson, T. (2006). From PIM to GIM: personal information management in group contexts. Commications of the ACM, 49, 1 (Jan. 2006), 74-75.

[9] Hawkins, D. & Wagers, R. (1982). Online bibliographic search strategy development. Online, 6(3), 12-19

[10] Jones, W., Bruce, H., & Dumais, S. (2001). Keeping found things found on the Web. Proceedings of the 2001 ACM CIKM 10th International Conference on Information and Knowledge Management (Atlanta, GA, November 5-10, 2001) New York: Association for Computing Machinery, 119-134.

[11] Komlodi, A. (2002). Search history for user support in information-seeking interfaces. Unpublished doctoral dissertation, University of Maryland, College Park.

[12] Komlodi, A., Marchionini, G., and Soergel, D. (2007). Search history support for finding and using information: user interface design recommendations from a user study. Inf. Process. Manage. 43, 1 (Jan. 2007), 10-29.

[13] Marchionini, G., Haas, S., Plaisant, C., Shneiderman, B., & Hert, C. (2003). Toward a statistical knowledge network. Proceedings of the Third Conference on Digital Government Research dg.03 (Boston, May 18-21, 2003). P. 27-32.

[14] Marchionini, G. and Brunk, B. (2003). Towards a general relation browser: A GUI for information architects. Journal of Digital Information, 4(1).

[15] Shah, C. & Marchionini, G. (2007). Preserving 2008 US Presidential Election Videos. Paper at the 7th International Workshop on Web Archiving and Digital Preservation (IWAW'07). http://www.ils.unc.edu/vidarch/Shah-IWAW07.pdf

[16] Shen, X., Tan, B., & Zhai, C. (2005a).UCAIR: Capturing and exploiting context for personalized search. In Proceedings of 2005 ACM Conference on Research and Development on Information Retrieval - Information Retrieval in Context Workshop (IRiX'2005).

[17] Shen, X., Tan, B., & Zhai, C. (2005b). Context-sensitive information retrieval using implicit feedback. In Proceedings of 2005 ACM Conference on Research and Development on Information Retrieval (SIGIR '05), 43-50.

[18] Shneiderman, B. (1994). Dynamic Queries for Visual Information Seeking, IEEE Software, 11(6): 70-77.

[19] Weinreich, H., Obendorf, H., Herder, E., & Mayer, M. (2008). Not quite the average: An empirical study of Web use. ACM Transactions on the Web, 2(1), Article #5.

[20] Whittaker, S.; Bellotti, V.; Gwizdka, J. (2006). Email as personal information management. Communications of the ACM. 2006 January; 49 (1): 68-73.

[21] Zhang, J. and Marchionini, G. (2005). Evaluation and evolution of a browse and search interface: relation browser. In Proceedings of the 2005 National Conference on Digital Government Research (Atlanta, Georgia, May 15 - 18, 2005). ACM International Conference Proceeding Series, vol. 89. Digital Government Research Center, 179-188.