

The Wrapper: An Open Source Application for Logging User – System Interactions during Searching Studies

Bernard J. Jansen

College of Information Sciences and Technology
The Pennsylvania State University
University Park, PA, 16801, USA
jjansen@ist.psu.edu

ABSTRACT

Conducting user studies and user evaluations of information retrieval systems requires the collection of specialized data, namely user interactions with the system. These interactions occur both on the server and the client side. Collecting the client-side data in a format for easy analysis has proven to be a difficult task for many researchers. In this paper, we report on a software application that logs user interactions with information searching systems. Developed in Visual Basic, the application is simple to install, ease to modify, and logs session actions to an ASCII text file for easy upload into a variety of analysis packages. The application records many of the typical user interactions with searching systems, including submitted queries, Web site navigation, scrolling, bookmarking, among others. The application time stamps each interaction. Interested researchers may download the application via a provided uniform resource locator.

Categories and Subject Descriptors

H.3.3 [1] Information Search and Retrieval – relevance feedback

General Terms

Performance, Design, Experimentation, Human Factors

Keywords

Implicit user feedback, search engine evaluation

1. INTRODUCTION

Client-side interactions can be extremely important in determining the underlying intent of the online information seeker. As such, implicit feedback actions are receiving increased attention from researchers as an additional information source to improve system design and content. These client-side interactions are also known as implicit feedback.

In prior work, Oard and Kim [12] classify types of implicit, and explicit, feedback along two axes, which are **Behavior Category** and **Minimum Scope**. The **Behavior Category** (*examine, retain, reference, and annotate*) refers to the purpose of the observed behavior. **Minimum Scope** (*segment, object, and class*) refers to the smallest possible scope of content to which the observed behavior could apply.

Continuing this line of research, Kelly and Teevan [11] provide a modification of Oard and Kim's classification, adding the classification of *create* to the **Behavior Category**. Both Oard

and Kim's [12] and Kelly and Teevan's [11] classifications include explicit actions, specifically behavior categories related to *create* and *annotate* of content. Additionally, the scope of implicit feedback presented by [11, 12] seems primarily focused on content.

Continuing this refining process, Table 1 presents implicit feedback actions related directly to information seeking within a hypermedia environment and implicit feedback on both Web information searching systems and content. In contrasting with prior work [11, 12] and building from previous work [4, 7], we narrow the **Behavior Category** to only implicit feedback and provide more specificity. We broaden the **Minimum Scope** beyond content to include also *System*.

We include (in Table 1) two additional **Behavior Categories** (*execute* and *navigate*). In addition to content, we incorporate *System* with a **Minimum Scope** of *Interface*. We further incorporate the actions of *open*, *close*, and *resize* (e.g., actions on a browser); the action of *click* (e.g., click on an URL in a results list); the actions of *create* and *name* (e.g., create a Favorites folder or name a Favorite); and the actions of *goto*, *previous*, and *next* (e.g., actions dealing with results lists).

Since Web searching work typically focuses fully on information searching, we have not included any information creation categories. Thus, the action of *create* applies to the *system* rather than the creation of content, as in [11, 12]. In this modified approach, there are additional implicit feedback actions beyond those dealing with content (i.e., implicit feedback concerning interactions with the system), the focus is specifically on searching for information rather than the creation and posting of information, and there are more implicit feedback actions included.

Most Web searching research has focused on the use of search engine transactions logs for data collection [c.f., 8, 9, 13, 14]. As a server side logging, transaction logs typically do not contain the full range of user – system interactions, especially implicit feedback that is typically client-side interactions. Therefore, researchers have to rely of other applications to log these interactions.

Hancock-Beaulieu, Robertson, and Nielsen [2] supplemented their transaction logging with an application that included online questionnaires. In their naturalistic and longitudinal study of professionals and their information seeking patterns, Choo, Betlor, and Turnbull [1] had to develop their own logging software.

Table 1. Classification of Implicit Feedback on System and Content [6]

		Minimum Scope			
		System	Content		
		Interface	Segment	Object	Class
Behavior Category	Execute	Query Open Close Resize	Click Scroll	Select	
	Examine		View Find	Open	Browse
	Navigate	Back Forward		GoTo Previous Next	
	Retain	Create Name	Print	Bookmark Save Purchase Email	
	Reference		Copy – Paste		

Kelly [10] used WinWhatWhere Investigator, which is a spy software package used to covertly “monitor” a person’s computer activities. Spy software has some inherent disadvantages for use in user studies and evaluation including granularity of data capture, and privacy concerns. Additionally, a proxy server supplemented data collection.

Toms, Freund, and Li [15] developed the WiIRE system for conducting large scale evaluations. This system facilitates the evaluation of dispersed study participants; however, it a server-side application focusing on the participant – interactions with Web server. As such, the entire “study” must occur within the WiIRE framework.

There are commercial applications for general-purpose (i.e., not specifically information retrieval (IR)) user studies. One such example of such a commercial software applications is Morae 1.1 (<http://www.techsmith.com/products/morae/default.asp>) offered by TechSmith. Morae provides extremely detailed tracking of user actions, including video capture over a network. However, Morae is not specifically tailored for IR studies and captures so much information at such a fine granularity that it significantly complicates data analysis process.

To assist in addressing this need, we have developed a software application for use in conjunction with transaction log analysis and other types of IR studies. The application is coded in a standard programming language (Visual Basic 6). It is ease to

install and collects a wide range of user – systems interactions. The application logs much of the user interactions identified by prior research [6, 11, 12], along with a the content of the interaction (i.e., URL, document, results listing, etc.). These implicit feedback actions and documents are referred to as action-object pairs [3]. We have validated the application in a series of user studies [5, 6] and have found the application to be extremely resilient, with near 100% operational effectiveness.

A description of the features and output of the application is presented, along with a uniform resource locator (URL) where interested researchers can download the application for use in their research projects and studies.

2. APPLICATION DESCRIPTION

The software application runs as an executable, generate from the Visual Basic programming environment. One can activate the application manually or via a bat file. The application has a window’s interface (Figure 1) for real time observation, which one can deactivate so that it does not display. The application logs interactions with the IR system, along with other applications, using Dynamic Data Exchange (DDE). Output is to a text file, with a specifiable location and an automatically generated unique filename.

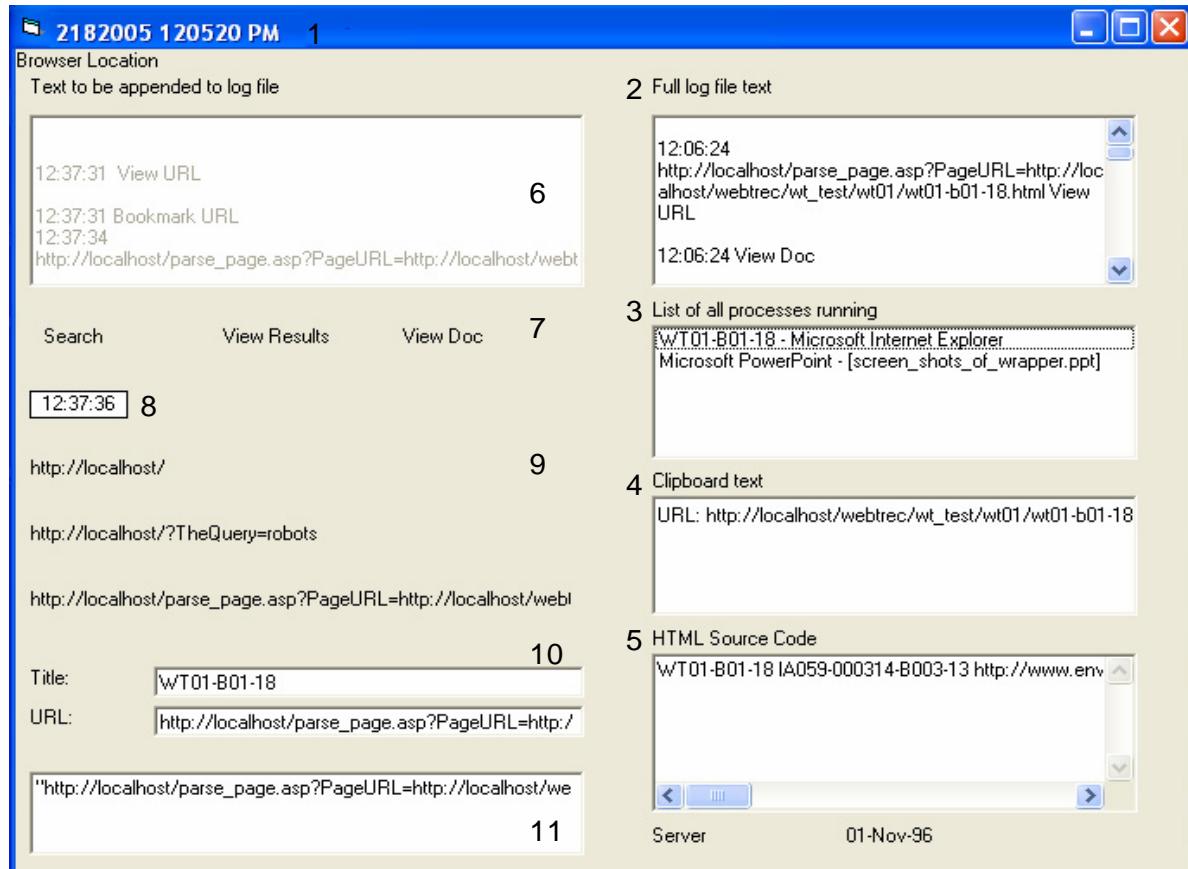


Figure 1. The Client-side Application with Action – Object Data Displayed.

Referring to Figure 1, we numbered each of the functional aspects of the application, which we describe below.

1. Log filename (generated automatically using date and time)
2. Running text of log file.
3. List of all processing running.
4. The current value of the clipboard.
5. HTML Source Code of the current page.
6. Text to be appended to log file.
7. The last three interactions logged.
8. Current system time.
9. Last three URLs visited.
10. Title and URL of current page.
11. Running list of URLs.

Table 2 shows an example of the application output.

Table 2: Transaction Log of User Interactions

Time Stamp	Interaction
12:12:44	http://localhost/
12:12:44	Search RON (Back Space) BOTS
12:12:56	http://localhost/?TheQuery=robots View URL
12:12:57	View Results
12:13:02	SCROLLED RESULTS
12:13:29	http://localhost/wt01/webtrec/wt01-b01-18.html
12:13:30	View Doc
12:13:34	SCROLLED PAGE

In its current version, the application logs a wide range of user interactions, include interactions with the browser tool bar, interactions with the system clipboard, scrolling of results listing or documents, and numerous implicit feedback actions [12], such as bookmark, copy, print, save, and scroll.

3. CONCLUSION

In this manuscript, we record on an open source application for use during user studies of IR systems. The application is focused on the typical interactions of searchers, thereby providing the needed granularity of data for fruitful analysis, without overwhelming amounts of data that slow the data analyze process. The

application is currently available for download at <http://ist.psu.edu/faculty/jansen/>. The application is currently available in two versions. Version A logs all interactions with the computer system, including all keyboard entries with all software applications. Version B only logs interactions with the browser.

In future research, we aim to increase the number of user interactions the application logs and enhance the application for naturalistic type studies, as well as increase the number of browsers that operate with the Wrapper. A client-server version (i.e., where the data is automatically sent from a client machine to a server) is in beta testing.

4. REFERENCE

- [1] Choo, C., Betlor, B., and Turnbull, D., A Behavioral Model of Information Seeking on the Web: Preliminary Results of a Study of How Managers and IT Specialists Use the Web, in *Proceedings of the 61st Annual Meeting of the American Society for Information Science*, 1998. Pittsburgh, PA. pp. 290-302.
- [2] Hancock-Beaulieu, M., Robertson, S., and Nielsen, C., Evaluation of online catalogues: an assessment of methods (BL Research Paper 78), The British Library Research and Development Department, London 1990.
- [3] Jansen, B. J., Designing Automated Help Using Searcher System Dialogues, in *Proceedings of the 2003 IEEE International Conference on Systems, Man & Cybernetics*, 2003. Washington, D.C., USA. 5-8 October. pp. 10 - 16.
- [4] Jansen, B. J., Seeking and Implementing Automated Assistance During the Search Process, *Information Processing and Management*, Forthcoming.
- [5] Jansen, B. J. and Kroner, G., The Impact of Automated Assistance on the Information Retrieval Process, in *Proceedings of The ACM CHI 2003 Conference on Human Factors in Computing Systems*, 2003. Fort Lauderdale, Florida. 5-10 April. pp. 1004 - 1006.
- [6] Jansen, B. J. and McNeese, M. D., Evaluating the Effectiveness of and Patterns of Interactions with Automated Searching Assistance, *Journal of the American Society for Information Science and Technology*, vol. 56, pp. 1480-1503, 2005.
- [7] Jansen, B. J. and Pooch, U., Assisting the Searcher: Utilizing Software Agents for Web Search Systems, *Internet Research - Electronic Networking Applications and Policy*, vol. 14, pp. 19-33, 2004.
- [8] Jansen, B. J., Spink, A., and Pederson, J., Trend Analysis of AltaVista Web Searching, *Journal of the American Society for Information Science and Technology*, vol. 56, pp. 559-570, 2005.
- [9] Jansen, B. J., Spink, A., and Saracevic, T., Real Life, Real Users, and Real Needs: A Study and Analysis of User Queries on the Web, *Information Processing & Management*, vol. 36, pp. 207-227, 2000.
- [10] Kelly, D., Understanding Implicit Feedback and Document Preference: A Naturalistic User Study, Rutgers, The State University of New Jersey, New Brunswick January 2004.
- [11] Kelly, D. and Teevan, J., Implicit Feedback for Inferring User Preference: A Bibliography, *SIGIR Forum*, vol. 37, pp. 18-28, 2003.
- [12] Oard, D. and Kim, J., Modeling Information Content Using Observable Behavior, in *Proceedings of the 64th Annual Meeting of the American Society for Information Science and Technology*, 2001. Washington, D.C., USA. 31 October - 4 November. pp. 38-45.
- [13] Park, S., Bae, H., and Lee, J., End User Searching: A Web Log Analysis of NAVER, a Korean Web Search Engine, *Library & Information Science Research*, vol. 27, pp. 203-221, 2005.
- [14] Silverstein, C., Henzinger, M., Marais, H., and Moricz, M., Analysis of a Very Large Web Search Engine Query Log, *SIGIR Forum*, vol. 33, pp. 6-12, 1999.
- [15] Toms, E. G., Freund, L., and Li, C., WiIRE: the Web interactive information retrieval experimentation system prototype, *Information Processing & Management*, vol. 40, pp. 655-675, 2004.