Chapter VI
The Methodology of Search Log Analysis

Bernard J. Jansen
Pennsylvania State University, USA

ABSTRACT

Exploiting the data stored in search logs of Web search engines, Intranets, and Websites can provide important insights into understanding the information searching tactics of online searchers. This understanding can inform information system design, interface development, and information architecture construction for content collections. This article presents a review of and foundation for conducting Web search transaction log analysis. A search log analysis methodology is outlined consisting of three stages (i.e., collection, preparation, and analysis). The three stages of the methodology are presented in detail with discussions of the goals, metrics, and processes at each stage. The critical terms in transaction log analysis for Web searching are defined. Suggestions are provided on ways to leverage the strengths and addressing the limitations of transaction log analysis for Web searching research.

INTRODUCTION

Information searching researchers have employed search logs for analyzing a variety of Web information systems (Croft, Cook, & Wilder, 1995; Jansen, Spink, & Saracevic, 2000; Jones, Cunningham, & McNab, 1998; Wang, Berry, & Yang, 2003). Web search engine companies use search logs (also referred to as transaction logs) to investigate searching trends and effects of system improvements (c.f., Google at http://www.google.com/press/zeitgeist.html or Yahoo! at http://buzz.yahoo.com/buzz_log/?fr=fp-buzz-morebuzz). Search logs are an unobtrusive method of collecting significant amounts of searching data on a sizable number of system users. There are several researchers who have employed the search log analysis methodology to study Web searching; however, not as many as one might expect.

One possible reason is that there are limited published works concerning how to employ search logs to support the study of Web searching, the use of Web search engines, Intranet searching, or other Web searching applications. None of the published works provide a comprehensive explanation of the methodology. This chapter addresses the use of search log analysis (also referred to as transaction log analysis) for the study of Web searching and Web search engines in order to facilitate its use as a research methodology. A three-stage process composed of data *collection, preparation, and analysis* is presented for transaction log analysis. Each stage is addressed in detail and a stepwise methodology to conduct transaction log analysis for the study of Web searching is described. The strengths and shortcomings of search log analysis are discussed.

REVIEW OF LITERATURE

What is a Search Log?

Not surprisingly, a search log is a file (i.e., log) of the communications (i.e., transactions) between a system and the users of that system. Rice and Borgman (1983) present transaction logs as a data collection method that automatically captures the type, content, or time of transactions made by a person from a terminal with that system. Peters (1993) views transaction logs as electronically recorded interactions between on-line information retrieval systems and the persons who search for the information found in those systems.

For Web searching, a search log is *an electronic record of interactions that have occurred during a searching episode between a Web search engine and users searching for information on that Web search engine*. A Web search engine may be a general-purpose search engine, a niche search engine, a searching application on a single Web site, or variations on these broad classifications. The users may be humans or computer programs acting on behalf of humans. Interactions are the communication exchanges that occur between users and the system. Either the user or the system may initiate elements of these exchanges.

How are These Interactions Collected?

The process of recording the data in the search log is relatively straightforward. Web servers record and store the interactions between searchers (i.e., actually Web browsers on a particular computer) and search engines in a log file (i.e., the transaction log) on the server using a software application. Thus, most search logs are server-side recordings of interactions. Major Web search engines execute millions of these interactions per day. The server software application can record various types of data and interactions depending on the file format that the server software supports.

Typical transaction log formats are access log, referrer log, or extended log. The W3C (http://www.w3.org/TR/WD-logfile.html) is one organizational body that defines transaction log formats. However, search logs are a special type of transaction log file. This search log format has most in common with the extended file format, which contains data such as the client computer's Internet Protocol (IP) address, user query, search engine access time, and referrer site, among other fields.

Why Collect This Data?

Once the server collects and records the data in a file, one must analyze this data in order to obtain beneficial information. The process of conducting this examination is referred to as *transaction log analysis* (TLA). TLA can focus on many interaction issues and research questions (Drott, 1998), but it typically addresses either issues of system performance, information structure, or user interactions.

In other views, Peters (1993) describes TLA as *the study of electronically recorded interactions between on-line information retrieval systems and* *the persons who search for information found in those systems*. Blecic and colleagues (1998) define TLA as the detailed and systematic examination of each search command or query by a user and the following database result or output. Phippen, Shepherd, and Furnell (2004) and Spink and Jansen (2004) also provide comparable definitions of TLA.

For Web searching research, we focus on a sub-set of TLA, namely search log analysis (SLA). One can use TLA to analyze the browsing or navigation patterns within a Website, while SLA is concerned exclusively with searching behaviors. SLA is defined as *the use of data collected in a search log to investigate particular research questions concerning interactions among Web users, the Web search engine, or the Web content during searching episodes*. Within this interaction context, SLA could use the data in search logs to discern attributes of the search process, such as the searcher's actions on the system, the system responses, or the evaluation of results by the searcher.

The goal of SLA is to gain a clearer understanding of the interactions among searcher, content and system or the interactions between two of these structural elements, based on whatever research questions are the drivers for the study. From this understanding, one achieves some stated objective, such as improved system design, advanced searching assistance, or better understanding of some user information searching behavior.

What is the Theoretical Basis of TLA (and SLA)?

TLA and its sub-component, SLA, lend themselves to a grounded theory approach (Glaser & Strauss, 1967). This approach emphasizes a systematic discovery of theory from data using methods of comparison and sampling. The resulting theories or models are grounded in observations of the "real world," rather than being abstractly generated. Therefore, grounded theory is an inductive approach to theory or model development, rather than the deductive alternative (Chamberlain, 1995).

Using SLA as a methodology in information searching, one examines the characteristics of searching episodes in order to isolate trends and identify typical interactions between searchers and the system. Interaction has several meanings in information searching, addressing a variety of transactions including query submission, query modification, results list viewing, and use of information objects (e.g., Web page, pdf file, video). Efthimiadis and Robertson (1989) categorize interaction at various stages in the information retrieval process by drawing from information-seeking research. SLA deals with the tangible interaction between user and system in each of these stages. SLA addresses levels one and two (*move* and *tactic*) of Bates' (1990) four levels of interaction, which are *move*, *tactic*, *stratagem*, and *strategy*. Belkin and fellow researchers (1995) have extensively explored user interaction based on user needs, from which they developed a multi-level view of searcher interactions. SLA focuses on the specific expressions of these user needs. Saracevic (1997) views interaction as the exchange of information between users and system. Increases in interaction result from increases in communication content. SLA is concerned with the exchanges and manner of these exchanges. Hancock-Beaulieu (2000) identifies three aspects of interaction, which are interaction within and across tasks, interaction as task sharing, and interaction as a discourse. One can use SLA to analyze the interactions within, across, and sharing.

For the purposes of SLA, interactions can be considered *the physical expressions of communication exchanges between the searcher and the system*. For example, a searcher may submit a query (i.e., an interaction). The system may respond with a results page (i.e., an interaction). The searcher may click on a uniform resource locator (URL) in the results listing (i.e., an interaction). Therefore, for SLA, interaction is a more mechanical expression of underlying information needs or motivations.

How is SLA Used?

Researchers and practitioners have used SLA (usually referred to as TLA in these studies) to evaluate library systems, traditional information retrieval (IR) systems, and more recently Web systems. Transaction logs have been used for many types of analysis; in this review, we focus on those studies that centered on or about searching. Peters (1993) provides a review of TLA in library and experimental IR systems. Some progress has been made in TLA methods since Peters' summary (1993) in terms of collection and ability to analyze data. Jansen and Pooch (2001) report on a variety of studies employing TLA for the study of Web search engines and searching on Web sites. Jansen and Spink (2005) provide a comprehensive review of Web searching TLA studies. Other review articles include Kinsella and Bryant (1987) and Fourie (2002).

Employing TLA in research projects, Meister and Sullivan (1967) may be the first to have conducted and documented TLA results, and Penniman (1975) appears to have published one of the first research articles using TLA. There have been a variety of TLA studies since (c.f., Baeza-Yates & Castillo, 2001; Chau, Fang, & Sheng, 2006; Fourie & van den Berg, 2003; Millsap & Ferl, 1993; Moukdad & Large, 2001; Park, Bae, & Lee, 2005).

Several papers have discussed the use of TLA as a methodological approach. Sandore and Kaske (1993) review methods of applying the results of TLA. Borgman, Hirsch, and Hiller (1996) comprehensively review past literature to identify the methodologies that these studies employed, including the goals of the studies. Several researchers have viewed TLA as a high-level designed process, including Copper (1998). Other researchers, such as Hancock-Beaulieu, Robertson, and Nielsen (1990), Griffiths, Hartley, and Willson (2002), Bains (1997), Hargittai (2002), and Yuan and Meadows (1999), have advocated using TLA in conjunction with other research methodologies or data collection. Alternatives for other data collection include questionnaires, interviews, video analysis, and verbal protocol analysis.

How is SLA Critiqued?

Almost from its first use, researchers have critiqued TLA as a research methodology (Blecic et al., 1998; Hancock-Beaulieu et al., 1990; Phippen et al., 2004). These critiques report that transaction logs do not record the users' perceptions of the search, cannot measure the underlying information need of the searchers, and cannot gauge the searchers' satisfaction with search results. In this vein, Kurth (1993) reports that transaction logs can only deal with the actions that the user takes, not their perceptions, emotions, or background skills.

Kurth (1993) further identifies three methodological issues with TLA, which are: *execution*, *conception*, and *communication*. Kurth (1993) states that TLA can be difficult to execute due to collection, storage, and analysis issues associated with the hefty volume and complexity of the dataset (i.e., significant number of variables). With complex datasets, it is sometimes difficult to develop a conceptual methodology for analyzing the dependent variables. Communication problems occur when researchers do not define terms and metrics in sufficient detail to allow other researchers to interpret and verify their results.

Certainly, any researcher who has used TLA would agree with these critiques. However, upon reflection, these are issues with many, if not all, empirical methodologies (McGrath, 1994). Further, although Kurth's critique (1993) is still generally valid, advances in transaction logging software, standardized transaction log formats, and improved data analysis software and methods have addressed many of these shortcomings.

Certainly, the issue with terms and metrics still apply (Jansen & Pooch, 2001).

As an additional limitation, transaction logs are primarily a server-side data collection method; therefore, some interaction events (Hilbert & Redmiles, 2001) are masked from these logging mechanisms, such as when the user clicks on the *back* or *print* button on the browser software, or *cuts* or *pastes* information from one window to another on a client computer. Transaction logs also, as stated previously, do not record the underlying situational, cognitive, or affective elements of the searching process, although the collection of such data can inform system design (Hilbert & Redmiles, 1998).

What are the Tools to Support SLA?

In an effort to address these issues, Hancock-Beaulieu, Robertson, and Nielsen (1990) developed a transaction logging software package that included online questionnaires to enhance TLA of browsing behaviors. This application was able to gather searcher responses via the questionnaires, but it also took away the unobtrusiveness (one of the strengths of the method) of the transaction log approach. Some software has been developed for unobtrusively logging client-side types of events, for example, the *Tracker* research package (Choo, Betlor, & Turnbull, 1998; Choo & Turnbull, 2000), the Wrapper (Jansen, Ramadoss, Zhang, & Zang, 2006), and commercial spyware software systems.

In other tools for examining transaction log data, Wu, Yu, and Ballman (1998) present SpeedTracer, which is a tool for data mining Web server logs. However, given that transaction log data is usually stored in ASCII text files, relational databases or text-processing scripts work extremely well for TLA. Wang, Berry, and Yang (2003) used a relational database, as did Jansen, Spink, and Saracevic (2000) and Jansen, Spink, and Pederson (2005). Silverstein, Henzinger, Marais, and Moricz (1999) used text processing scripts. All approaches have advantages and disadvantages. With text processing scripts, the analysis can be done in one pass. However, if additional analysis needs to be done, the whole dataset must be reanalyzed. With the relational database approach, the analysis is done in incremental portions; and one can easily add additional analysis steps, building from what has already been done.

In another naturalistic study, Kelly (2004) used WinWhatWhere Investigator, which is a spy software package that covertly "monitors" 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. Toms, Freund, and Li (2004) developed the WiIRE system for conducting large scale evaluations. This system facilities the evaluation of dispersed study participants; however, it is a server-side application focusing on the participant's interactions with the Web server. As such, the entire "study" must occur within the WiIRE framework.

There are commercial applications for general purpose (i.e., not specifically IR) user studies. An example 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 information searching studies and captures so much information at such a fine granularity that it significantly complicates the data analysis process.

How to Conduct TLA for Web Searching Research?

Despite the abundant literature on TLA, there are few published manuscripts on how actually to conduct it, especially with respect to SLA for Web searching. Some works do provide fairly comprehensive descriptions of the methods employed including Cooper (1998), Nicholas, Hunteytenn, and Lievestey (1999), Wang, Berry, and Yang (2003), and Spink and Jansen (2004). However, none of these articles presents a process or procedure for actually conducting TLA in sufficient detail to replicate the method. This chapter attempts to address this shortcoming building on work presented in (Jansen, 2006).

SLA PROCESS

Naturally, research questions need to be articulated to determine what data needs to be collected. However, search logs are typically of standard formats due to previously developed software applications. Given the interactions between users and Web browsers, which are the interfaces to Web search engines, the type of data that one can collect is standard. Therefore, the SLA methodology provided in this chapter is applicable to a wide range of studies.

SLA involves the following three major stages, which are:

- **Data Collection:** The process of collecting the interaction data for a given period in a transaction log;
- **Preparation:** The process of cleaning and preparing the transaction log data for analysis; and
- **Analysis:** The process of analyzing the prepared data.

Data Collection

The research questions define what information one must collect in a search log. Transaction logs provide a good balance between collecting a robust set of data and unobtrusively collecting that data (McGrath, 1994). Collecting data from real users pursuing needed information while interacting with real systems on the Web affects the type of data that one can realistically assemble. If one is conducting a naturalistic study (i.e., outside of the laboratory) on a real system (i.e., a system used by actual searchers), the method of data monitoring and collecting should not interfere with the information searching process. In addition to the loss of potential customers, a data collection method that interferes with the information searching process may unintentionally alter that process.

Fields in a Standard Search Log

Table 1 provides a sample of a standard search log format collected by a Web search engine.

The fields are common in standard Web search engine logs, although some systems may log additional fields. A common additional field is a cookie identification code that facilitates identifying individual searchers using a common computer. A cookie is a text message given by a Web server to a Web browser. The cookie is stored on the client machine.

In order to facilitate valid comparisons and contrasts with other analysis, a standard terminology and set of metrics (Jansen & Pooch, 2001) is advocated. This standardization will help address one of Kurth's critiques (1993) concerning the communication of SLA results across studies. Others have also noted terminology as an issue in Web research (Pitkow, 1997). The standard field labels and descriptors are presented below.

A *searching episode* is a series of searching interactions within a given temporal span by a single searcher. Each record, shown as a row in Table 1, is a *searching interaction*. The format of each *searching interaction* is:

- *User Identification*: The IP address of the client's computer. This is sometimes also an anonymous user code address assigned by the search engine server, which is our example in Table 1.
- *Date:* The date of the interaction as recorded by the search engine server.
- *The Time*: The time of the interaction as recorded by the search engine server.

Table 1. Snippet from a Web Search Engine Search Log

user identification	**date**	**thetime**	**search_url**
ce00160c04c4158087704275d69fbecd	25/Apr/2004	04:08:50	Sphagnum Moss Harvesting + New Jersey + Raking
38f04d74e651137587e9ba3f4f1af315	25/Apr/2004	04:08:50	emailanywhere
fabc953fe31996a0877732a1a970250a	25/Apr/2004	04:08:54	Tailpiece
5010dbbd750256bf4a2c3c77fb7f95c4	25/Apr/2004	04:08:54	1'personalities AND gender AND education'1
25/Apr/2004	**04:08:54**	**dmr panasonic**	
89bf2acc4b64e4570b89190f7694b301	25/Apr/2004	04:08:55	bawdy poems"
	"Mark Twain""	**25/Apr/2004**	
397e056655f01380cf181835dfc39426		**04:08:56**	**gay porn**
a9560248d1d8d7975ffc455fc921cdf6	25/Apr/2004	04:08:58	skin diagnostic
81347ea595323a15b18c08ba5167fbe3	25/Apr/2004	04:08:59	Pink Floyd cd label cover scans
3c5c399d3d7097d3d01aeea064305484	25/Apr/2004	04:09:00	freie stellen dangaard
9dafd20894b6d5f156846b56cd574f8d	25/Apr/2004	04:09:00	Moto.it
415154843dfe18f978ab6c63551f7c86	25/Apr/2004	04:09:00	Capablity Maturity Model VS.
c03488704a64d981e263e3e8cf1211ef	25/Apr/2004	04:09:01	ana cleonides paulo fontoura

Note: Bolded items are intentional errors

- *Search URL*: The query terms as entered by the user.

Web search engine server software normally always records these fields. Other common fields include *Results Page* (a code representing a set of result abstracts and URLs returned by the search engine in response to a query), *Language* (the user preferred language of the retrieved Web pages), *Source* (the federated content collection searched, also known as *Vertical*), and *Page Viewed* (the URL that the searcher visited after entering the query and viewing the results page, which is also known as *click-thru or click-through*).

Data Preparation

Once the data is collected, one moves to the data preparation stage of the SLA process. For data preparation, the focus is on importing the search log data into a relational database (or other analysis software), assigning each record a primary key, cleaning the data (i.e., checking each field for bad data), and calculating standard interaction metrics that will serve as the basis for further analysis.

Figure 1 shows the Entity – Relation (ER) diagram for the relational database that will be used to store and analyze the data from our search log.

An ER diagram models the concepts and perceptions of the data and displays the conceptual schema for the database using standard ER notation. Table 2 presents the legend for the schema constructs names.

Since search logs are in ASCII format, one can easily import the data into most relational databases. A key thing is to import the data in the same coding schema in which it was recorded (e.g., UTF-8, US-ASCII). Once imported, each record is assigned a unique identifier or primary key. Most modern databases can assign this au-

Figure 1. ER Scheme Diagram Web Search Log

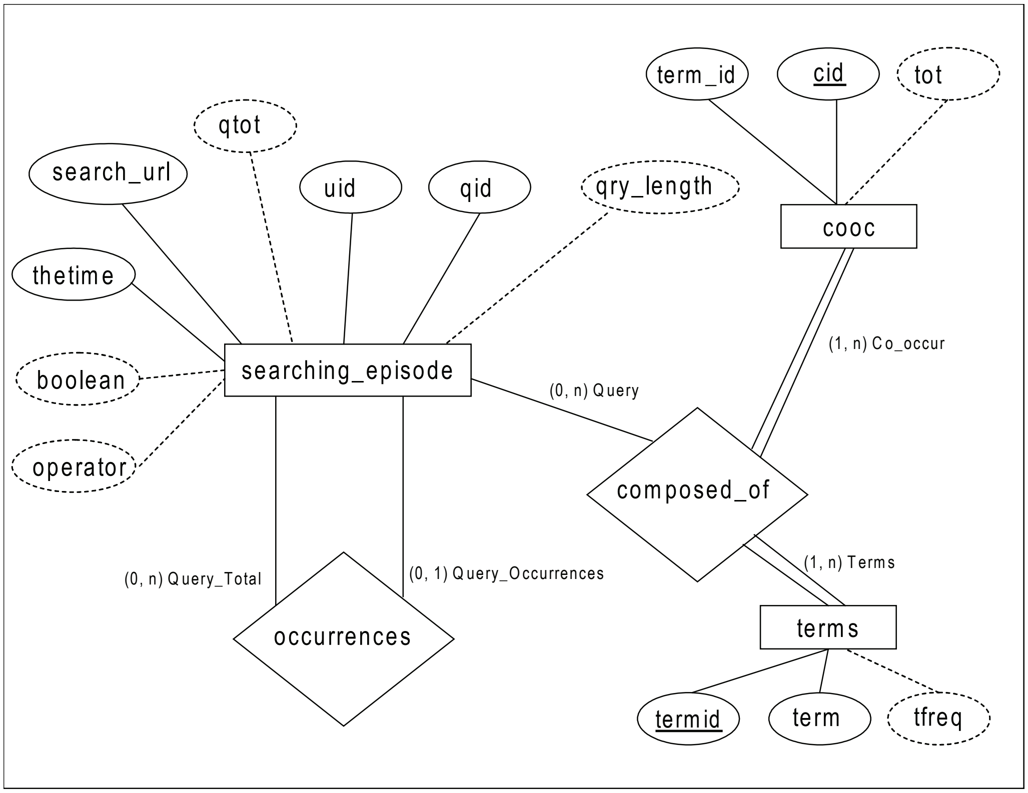

Table 2. Legend for ER Schema Constructs for Search Log.

Entity Name	**Construct**
Searching_Episodes	a table containing the searching interactions
boolean	denotes if the query contains Boolean operators
operators	denotes if the query contains advanced query operators
q_length	query length in terms
qid	primary key for each record
qtot	number of results pages viewed
searcher_url	query terms as entered by the searcher
thetime	time of day as measured by the server
uid	user identification based on IP
Terms	table with terms and frequency
term_ID	term identification
term	term from the query set
tfreq	number of occurrences of term in the query set
Cooc	table term pairs and the number of occurrences of those pairs
term_ID	term identification
cid	the combined term identification for a pair of terms
tot	number of occurrences of the pair in the query set

tomatically on importation, or one can assign it later using scripts.

Cleaning the Data

Once the search log data is in a suitable analysis software package, the focus shifts to cleaning the data. Records in search logs can contain corrupted data. These corrupted records can be as a result of multiple reasons; but they are mostly related to errors when logging the data. In the example shown in Table 1, one can easily spot these records (additionally these records are bolded), but many times a search log will number millions if not billions of records. Therefore, a visual inspection is not practical for error identification. From experience, one method of rapidly identifying most errors is to sort each field in sequence. Since the erroneous data will not fit the pattern of the other data in the field, these errors will usually appear at the top of, bottom of, or grouped together in each sorted field. Standard database functions to sum and group key fields such as time and IP address will usually identify any further errors. One must remove all records with corrupted data from the transaction log database. Typically, the percentage of corrupted data is small relative to the overall database.

Parsing the Data

Using the three fields of *The Time*, *User Identification*, and *Search URL*, common to all Web search logs, the chronological series of actions in a searching episode is recreated. The Web query search logs usually contain queries from both human users and agents. Depending on the research objective, one may be interested in only individual human interactions, those from common user terminals, or those from agents. For the running example used in this chapter, we will consider the case of only having an interest in human searching episodes. To do this, all sessions with less than 101 queries are separated into an individual search log for this example.

Given that there is no way to accurately identify human from non-human searchers (Silverstein et al., 1999; Sullivan, 2001), most researchers using Web search log either ignore it (Cacheda & Viña, 2001) or assume some temporal or interaction cut-off (Montgomery & Faloutsos, 2001; Silverstein et al., 1999). Using a cut-off of 101 queries, the subset of the search log is weighted to queries submitted primarily by human searchers in a non-common user terminal, but 101 queries is high enough not to introduce bias by too low of a cut-off threshold. The selection of 101 is arbitrary, and other researchers have used a wide variety of cut-offs.

There are several methods to remove these large sessions. One can code a program to count the session lengths and then delete all sessions that have lengths over 100. For smaller log files (a few million or so records), it is just as easy to do with SQL queries. To do this, one must first remove records that do not contain queries. From experience, search logs may contain many such records (usually on the order of 35 to 40 percent of all records) as users go to Web sites for purposes other than searching.

Normalizing Searching Episodes

When a searcher submits a query, then views a document, and returns to the search engine, the Web server typically logs this second visit with the identical user identification and query, but with a new time (i.e., the time of the second visit). This is beneficial information in determining how many of the retrieved *results page*s the searcher visited from the search engine, but unfortunately, it also skews the results in analyzing how the query level of analysis. In order to normalize the searching episodes, one must first separate these result page requests from query submissions for each searching episode. An example of how to do this can be found in the SQL query #00 (Appendix A).

From a *tbl_main*, this will create a new table *tbl_searching_episodes* which contains a count of multiple submissions (i.e., *qtot*) from each searcher within each record as shown in Figure 2. This collapses the search log by combining all identical queries submitted by the same user to give the unique queries in order to analyze sessions, queries and terms, and pages of results (i.e., *tbl_searching_episodes*). Use the complete un-collapsed sessions (i.e., *tbl_main*) in order to obtain an accurate measure of the temporal length of sessions. The *tbl_searching_episodes* will now be used for the remainder of our TLA. Use SQL query #01, Appendix A to identify the sessions with more than 100 records. Then, one can delete these records from *tbl_searching_episodes* using the SQL delete query #02, Appendix A.

In SLA, many times one is interested in terms and term usage, which can be an entire study in itself. In these cases, it is often cleaner to generate separate tables that contain each term and their frequency of occurrence. A term co-occurrence table that contains each term and its co-occurrence with other terms is also valuable for understanding the data. If using a relational database, one can generate these tables using scripts. If using text-parsing languages, one can parse these terms and associated data out during initial processing.

Figure 2. Records of Searching Episodes with Number of Duplicate Queries (qtot) Recorded

Microsoft Access - [tbl_searching_episodes : Table]

qid	uid	date	thetime	search_url	qtot	qry_length	boolean	operator
1	ce00160c04c4158087704275d69fbecd	25/Apr/2004	04:08:50	Sphagnum Moss Harvesting + New Jersey + Raking	4	6	☐	☑
2	38f04d74e651137587e9ba3f4f1af315	25/Apr/2004	04:08:50	emailanywhere	2	1	☐	☐
3	fabc953fe31996a0877732a1a970250a	25/Apr/2004	04:08:54	Tailpiece	1	1	☐	☐
4	5010dbbd750256bf4a2c3c77fb7f95c4	25/Apr/2004	04:08:54	1'personalities AND gender AND education'1	1	5	☑	☐
5	daade90d883432ddf6cf3fb509e3bf41	25/Apr/2004	04:08:54	dmr panasonic	1	2	☐	☐
6	89bf2acc4b64e4570b89190f7694b301	25/Apr/2004	04:08:55	bawdy poems"	1	2	☐	☑
7	9f6fa2a8d5b12a1f8380ed4ea1483b2b	25/Apr/2004	04:08:56	"Mark Twain""	1	2	☐	☑
8	397e056655f01380cf181835dfc39426	25/Apr/2004	04:08:56	gay porn	1	2	☐	☐
9	a9560248d1d8d7975ffc455fc921cdf6	25/Apr/2004	04:08:58	skin diagnostic	1	2	☐	☐
10	81347ea595323a15b18c08ba5167fbe3	25/Apr/2004	04:08:59	Pink Floyd cd label cover scans	1	6	☐	☐
11	3c5c399d3d7097d3d01aeea064305484	25/Apr/2004	04:09:00	freie stellen dangaard	1	3	☐	☐
12	9dafd20894b6d5f156846b56cd574f8d	25/Apr/2004	04:09:00	Moto.it	1	1	☐	☐
13	415154843dfe18f978ab6c63551f7c86	25/Apr/2004	04:09:00	Capablity Maturity Model VS.	1	4	☐	☐
14	c03488704a64d981e263e3e8cf1211ef	25/Apr/2004	04:09:01	ana cleonides paulo fontoura	1	4	☐	☐
15	7ab6a96ee504b5f54773f8cff3999354	25/Apr/2004	04:09:01	quetschrippen konstruktion kunststoff	2	3	☐	☐
16	01305f695166f2fc135bf26ee52d2a3b	25/Apr/2004	04:09:02	lovette password	3	2	☐	☐
17	eedec6d2ecc3519ea7747db9982abf57	25/Apr/2004	04:09:04	free porn	1	2	☐	☐
18	3438f445fa250f7377befde4f188a09b	25/Apr/2004	04:09:04	centro	2	1	☐	☐
19	ef8871c24d88b96ee8813cb361445040	25/Apr/2004	04:09:07	sex toys	1	2	☐	☐
20	cbf06662b4977b1b3161f1f3ea7e4ae4	25/Apr/2004	04:03:18	news "blue oak" "Quercus douglas"	1	5	☐	☑
21	9a44e9c5502a6b9ba9af1f0e443245b9	25/Apr/2004	04:09:09	international investment on nanotechnology	1	4	☐	☐
22	e40dab084acb6776a3aa57e8cea5b82f	25/Apr/2004	04:09:10	mosquito keychain	2	2	☐	☐
23	fabc953fe31996a0877732a1a970250a	25/Apr/2004	04:09:12	Valve Tailpiece	1	2	☐	☐
24	74574c974f7301ea4ba056fa00fab71b	25/Apr/2004	04:09:13	" wayne frocklage" Canadian pacific railway	1	5	☐	☑
25	143e559163fbf75e16a52a6d550fb01c	25/Apr/2004	04:02:46	matcho central	1	2	☐	☐
26	fc66620a68627983d416a8dd1afa0d73	25/Apr/2004	04:02:46	Pascal convert decimal to hex	2	5	☐	☐
27	7bff0819a1cdaf645f27f39842bde4b5	25/Apr/2004	04:02:46	bang bros"king chile"	1	4	☐	☑
28	59515ed43dfaa2dde796f3bab8045514	25/Apr/2004	04:02:46	CFISCO	1	1	☐	☐
29	3fb5e9a52e76f0fc344977c27e914ea8	25/Apr/2004	04:02:47	"Stargate" desktop themes	1	3	☐	☑
30	dacca5152e8e0975629cb994fbc0b447	25/Apr/2004	04:02:51	carrieland gallery	1	2	☐	☐
31	4167eb02168d2bc169ffa196d7b10fd5	25/Apr/2004	04:02:53	"femdom picture"	1	2	☐	☑
32	cd7b9dc409c60de204b94105830176ba	25/Apr/2004	04:02:55	"Carla Tonetti"	1	2	☐	☑
33	6b19f010e49a39d7103eabba4530fb74	25/Apr/2004	04:02:56	efficacy	1	1	☐	☐
34	5a20077ed4f1d8a7b41de52aca6ad756	25/Apr/2004	04:02:58	http://kickme.to/xaliasx	1	1	☐	☐
35	93cde4db94cc12cc852e7c4ad352620d	25/Apr/2004	04:03:14	P.M."	1	1	☐	☑
36	1c1c9d058d8dd3cfadb3ee54851f3624	25/Apr/2004	04:03:01	quattro elementi	1	2	☐	☐
37	bae5567110a17a941254661fda3205f0	25/Apr/2004	04:03:02	measurement high temp	1	3	☐	☐
38	606eb104503f078f5fdf20385e58544a	25/Apr/2004	04:03:02	levicom watercube	2	2	☐	☐
39	0fcec5c427a7a1b9393f538ff5373904	25/Apr/2004	04:03:03	notebook mediamarkt	1	2	☐	☐
40	082206ccd56dddc56894304522400d4d	25/Apr/2004	04:03:03	www.dancentury.com/cicada/wedding.html	1	1	☐	☐
41	5b43061638aef9d59dd351542c98f1fa	25/Apr/2004	04:03:04	mensajes subliminales un la publicidad"	1	5	☐	☑
42	07ff54fb225084bfe3c746c320222b5b	25/Apr/2004	04:03:13	online texting	1	2	☐	☐
43	07ff54fb225084bfe3c746c320222b5b	25/Apr/2004	04:03:04	onlinetexting	1	1	☐	☐
44	177219518313a13d3469e8ad9d1d5a7c	25/Apr/2004	04:03:12	Art	1	1	☐	☐
45	177219518313a13d3469e8ad9d1d5a7c	25/Apr/2004	04:03:14	Art history	1	1	☐	☐
46	cd7b9dc409c60de204b94105830176ba	25/Apr/2004	04:02:57	"Jim Jansen"	1	2	☐	☑
*						0	☐	☐

Record: 1 of 46

Datasheet View

We see these as *tbl_terms* and *tbl_cooc* in our database (see Figure 1 and Table 2).

There are already several fields in our database, many of which can provide valuable information (see Figure 1 and Table 2). From these items, one can calculate several metrics, some of which take a long time to compute for large datasets.

DATA ANALYSIS

This stage focuses on three levels of analysis. These levels are discussed and the data analysis stage is stepped through.

Analysis Levels

The three common levels of analysis for examining transaction logs are *term*, *query, and session.*

Term Level Analysis

The term level of analysis naturally uses the *term* as the basis for analysis. A *term* is a string of characters separated by some delimiter such as a space or some other separator. At this level of analysis, one focuses on measures such as *term occurrence*, which is the frequency that a particular term occurs in the transaction log. *Total terms* is the number of terms in the dataset. *Unique terms* are the terms that appear in the data regardless of the number of times they occur. *High Usage Terms* are those terms that occur most frequently in the dataset. *Term co-occurrence* measures the occurrence of term pairs within queries in the entire search log. One can also calculate degrees of association of term pairs using various statistical measures (c.f., Ross & Wolfram, 2000; Silverstein et al., 1999; Wang et al., 2003).

The mutual information formula measures term association and does not assume mutual independence of the terms within the pair. We calculate the mutual information statistic for all term pairs within the data set. Many times, a relatively low frequency term pair may be strongly associated (i.e., if the two terms always occur together). The mutual information statistic identifies the strength of this association. The mutual information formula used in this research is:

$$I(w_1, w_2) = \ln \frac{P(w_1, w_2)}{P(w_1)P(w_2)}$$

where $P(w_1)$, $P(w_2)$ are probabilities estimated by relative frequencies of the two words and $P(w_1, w_2)$ is the relative frequency of the word pair and order is not considered. Relative frequencies are observed frequencies (F) normalized by the number of the queries:

$$P(w_1) = \frac{F_1}{Q'}; P(w_1) = \frac{F_2}{Q'}; P(w_1, w_2) = \frac{F_{12}}{Q'}$$

Both the frequency of term occurrence and the frequency of term pairs are the occurrence of the term or term pair within the set of queries. However, since a one term query cannot have a term pair, the set of queries for the frequency base differs. The number of queries for the terms is the number of non-duplicate queries in the data set. The number of queries for term pairs is defined as:

$$Q' = \sum_{n}^{m} (2n-3)Q_n$$

where Q_n is the number of queries with n words ($n > 1$), and m is the maximum query length. So, queries of length one have no pairs. Queries of length two have one pair. Queries of length three have three possible pairs. Queries of length four have five possible pairs. This continues up to the queries of maximum length in the data set. The formula for queries of term pairs (Q') account for this term pairing.

Query Level Analysis

The query level of analysis uses the query as the base metric. A *query* is defined as a string list of one or more terms submitted to a search engine. This is a mechanical definition as opposed to an information searching definition (Korfhage, 1997). The first query by a particular searcher is the *initial query*. A subsequent query by the same searcher that is different than any of the searcher's other queries is a *modified query*. There can be several occurrences of different modified queries by a particular searcher. A subsequent query by the same searcher that is identical to one or more of the searcher's previous queries is an *identical query*.

In many Web search engine logs, when the searcher traverses to a new results page, this interaction is also logged as an *identical query*. In other logging systems, the application records the page rank. A results page is the list of results, either sponsored or organic (i.e., non-sponsored), returned by a Web search engine in response to a query. Using either *identical queries* or some results page field, one can analyze the result page viewing patterns of Web searchers.

One can examine other measures at the query level of analysis. A *unique query* refers to a query that is different from all other queries in the transaction log, regardless of the searcher. A *repeat query* is a query that appears more than once within the dataset by two or more searchers.

Query complexity examines the query syntax, including the use of advanced searching techniques such as Boolean and other query operators. *Failure rate* is a measure of the deviation of queries from the published rules of the search engine. The use of query syntax that the particular IR system does not support, but may be common on other IR systems, is *carry over*.

Session Level Analysis

At the session level of analysis, one primarily examines the within-session interactions (Hancock-Beaulieu, 2000). However, if the search log spanned more than one day or assigns some temporal limit to interactions from a particular user, one could examine between-sessions interactions. A *session interaction* is any specific exchange between the searcher and the system (i.e., submitting a query, clicking a hyperlink, etc.). A *searching episode* is defined as a series of interactions within a limited duration to address one or more information needs. This session duration is typically short, with Web researchers using between five and 120 minutes as a cutoff (c.f., He, Göker, & Harper, 2002; Jansen & Spink, 2003; Montgomery & Faloutsos, 2001; Silverstein et al., 1999). Each choice of time has an impact on the results, of course. The searcher may be multitasking (Miwa, 2001; Spink, 2004) within a searching episode, or the episode may be an instance of the searcher engaged in successive searching (Lin, 2002; Özmutlu, Özmutlu, & Spink, 2003; Spink, Wilson, Ellis, & Ford, 1998). This *session* definition is similar to the definition of a *unique visitor* used by commercial search engines and organizations to measure Web site traffic. The number of queries per searcher is the *session length*.

Session duration is the total time the user spent interacting with the search engine, including the time spent viewing the first and subsequent Web documents, except the final document. Session duration can therefore be measured from the time the user submits the first query until the user departs the search engine for the last time (i.e., does not return). This viewing time of the final Web document is not available since the Web search engine server does not record the time stamp. Naturally, the time between visits from the Web document to the search engine may not have been entirely spent viewing the Web document, which is a limitation of the measure.

A *Web document* is the Web page referenced by the URL on the search engine's results page. A Web document may be text or multimedia and, if viewed hierarchically, may contain a nearly unlimited number of sub-Web documents. A Web document may also contain URLs linking to other Web documents. From the results page, a searcher may click on a URL, (i.e., visit) one or more results from the listings on the result page. This is *click through analysis* and measures the page viewing behavior of Web searchers. One measures *document viewing duration* as the time from when a searcher clicks on a URL on a results page to the time that searcher returns to the search engine. Some researchers and practitioners refer to this type of analysis as *page view analysis*. *Click through analysis* is possible if the transaction log contains the appropriate data.

Conducting the Data Analysis

The key to successful SLA is conducting the analysis with an organized approach. One method is to sequentially number and label the queries (or coded modules) to correspond to the order of execution and to their function, since many of these queries must be executed in a certain order to obtain valid results. Many relational database management systems provide mechanisms to add descriptive properties to the queries. These can provide further explanations of the query function or relate these queries directly to research questions. Figure 3 illustrates the application of such an approach.

Figure 3. Sequentially numbered and descriptively labeled queries for SLA.

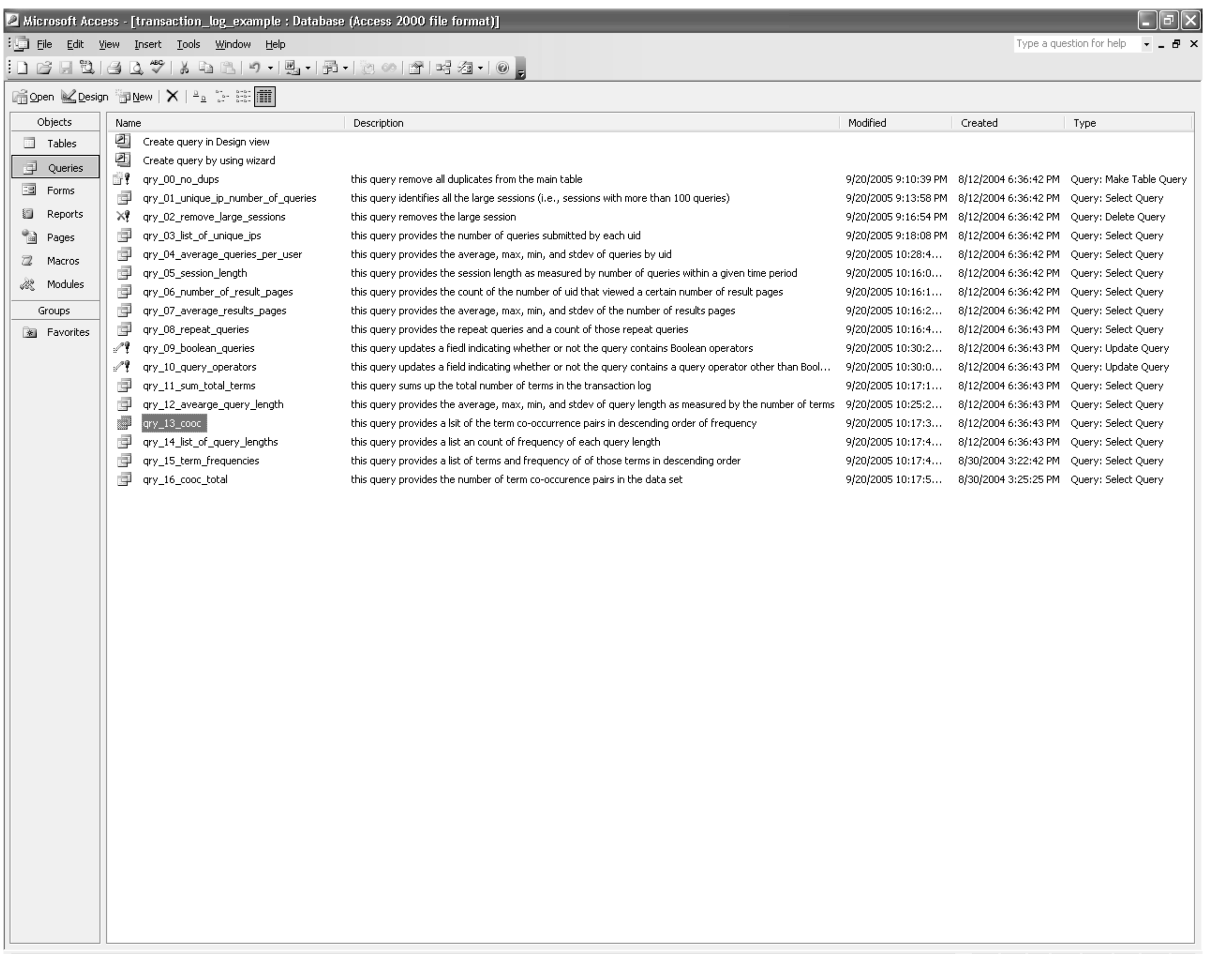

Figure 3 shows each query in sequence and provides a descriptive tag describing that query's function. To aid in reading, a list of queries is also provided in Appendix A.

One approaches SLA by conducting a series of standard analyses that are common to a wide variety of Web searching studies. Some of these analyses may directly address certain research questions. Others may be the basis for more in-depth research analysis.

One typical question is "How many searchers have visited the search engine during this period?" One can determine this by using SQL query 4, (Appendix A). This query will provide a list of unique searchers and the number of queries they have submitted during the period. One can modify this and determine "How many searchers have visited the search engine on each day during this period?" with the SQL query 5, Appendix A. Naturally, a variety of statistical results can be determined using the previous queries. For example, one can determine the standard deviation of number of queries per day using the SQL query #6, Appendix A.

One may want to know each of the session lengths (i.e., the number of queries within a session) for each searcher, which SQL query #7 will provide. Similarly, one may desire the number of searchers who viewed a certain number of results pages, addressed by query #8, Appendix A.

One can calculate various statistical results on results page viewing, such as the maximum number of result pages viewed using SQL query #10, Appendix A. SQL query #11, Appendix A will present the number of queries per day. An important aspect for system designers is results caching, because one needs to know the number of repeat queries submitted by the entire set of searchers during the period. The SQL query #12, Appendix A will tell us this information.

In order to understand how searchers are interacting with a search engine, the use of Boolean operators is an important feature. The SQL query #13, Appendix A makes a table of interactions with Boolean operators within the queries. Since most search engines offer other query syntax than just Boolean operators, the SQL query #14, Appendix A makes a table of queries containing other query syntax.

The SQL query #15, Appendix A provides a count of the number of terms within the transaction log. One certainly wants to know about query length; SQL query #16, Appendix A provides various statistics on query length: SQL query #17 provides the frequency of terms pairs within the transaction log, SQL query #18 provides a count of the various query lengths, SQL query #19 provides a count of the various term frequencies, and SQL query #20 provides a count of the term pairs within the transaction log.

The results from this series of queries both provides us a wealth of information about our data (e.g., occurrences of session lengths, occurrences of query length, occurrences of repeat queries, most used terms, most used term pairs) and serves as the basis for further investigations (e.g., session complexity, query structure, query modifications, term relationships).

DISCUSSION

It is certainly important to understand both the strengths and limitations of SLA for Web searching. First concerning the strengths, SLA provides a method of collecting data from a great number of users. Given the current nature of the Web, search logs appears to be a reasonable and non-intrusive means of collecting user - system interaction data during the Web information searching process from a large number of searchers. One can easily collect data on hundreds of thousands to millions of interactions, depending on the traffic of the Web site.

Second, one can collect this data inexpensively. The costs are the software and storage. Third, the data collection is unobtrusive, so the interactions represent the unaltered behavior of searchers, assuming the data is from an operational searching site. Finally, search logs are, at present, the only method for obtaining significant amounts of search data within the complex environment that is the Web (Dumais, 2002). Of course, researchers can also be doing SLA from research sites or capture client-side data across multiple sites using a custom Web browser (for the purpose of data collection) that does not completely mimic the searcher's natural environment.

There are limitations of SLA, as with any methodology. First, there may be certain types of data not in the transaction log, individuals' identities being the most common example. An IP address typically represents the "user" in a search log. Since more than one person may use a computer, an IP address is an imprecise representation of the user. Search engines are overcoming this limitation somewhat by the use of cookies.

Second, there is no way to collect demographic data when using search logs in a naturalistic setting. This constraint is true of many non-intrusive naturalistic studies. However, there are several sources for demographic data on the Web population based on observational and survey data. From these data sources, one may get reasonable estimations of needed demographic data. However, this still not attributable specific search data to specific sub-populations.

Third, a search log does not record the reasons for the search, the searcher motivations, or other qualitative aspects of use. This is certainly a limitation. In the instances where one needs this data, one should use transaction log analysis in conjunction with other data collection methods. However, this invasiveness then lessens the unobtrusiveness, which is an inherent advantage of search logs as a data collection method.

Fourth, the logged data may not be complete due to caching of server data on the client machine or proxy servers. This is an often mentioned limitation. In reality, this is a relatively minor concern for Web search engine research due to the method with which most search engines dynamically produce their results pages. For example, a user accesses the page of results from a search engine using the *Back* button of a browser. This navigation accesses the results page via the cache on the client machine. The Web server will not record this action. However, if the user clicks on any URL on that results page, functions coded on the results page redirects the click first to the Web server, from which the Web server records the visit to the Web site.

CONCLUSION

In this chapter, following the literature review, we presented a three-step methodology for conducting SLA, namely collecting, preparing, and analyzing. We then reviewed each step in detail, providing observations, guides, and lessons learned. We discussed the organization of the database at the ER-level, and we discussed the table design for standard search engine transaction logs. Furthermore, we provided 16 queries (Appendix B) one can use to conduct analysis. This presentation of the methodology at a detailed level of granularity will serve as an excellent basis for novice or experienced search log researchers.

Search logs are powerful tools for collecting data on the interactions between users and systems. Using this data, SLA can provide significant insights into user–system interactions, and it complements other methods of analysis by overcoming the limitations inherent in these methods. With respect to shortcomings, one can combine SLA with other data collection methods or other research results to improve the robustness of the analysis, when possible. Overall, SLA is a powerful tool for Web searching research, and the SLA process outlined here can be helpful in future Web searching research endeavors.

REFERENCES

Baeza-Yates, R., & Castillo, C. (2001, 1-5 May). *In Relating Web structure and user search behavior* (pp. 1-2). Paper presented at the 10th World Wide Web Conference, Hong Kong, China. ACM.

Bains, S. (1997). End-user searching behavior: Considering methodologies. *The Katharine Sharp Review,* 1(4), http://www.lis.uiuc.edu/review/winter1997/bains.html.

Bates, M.J. (1990). Where should the person stop and the information search interface start? *Information Processing & Management,* 26(5), 575-591.

Belkin, N., Cool, C., Stein, A., & Theil, S. (1995). Cases, scripts, and information-seeking strategies: On the design of interactive information retrieval systems. *Expert Systems With Applications,* 9(3), 379-395.

Blecic, D., Bangalore, N.S., Dorsch, J.L., Henderson, C.L., Koenig, M.H., & Weller, A.C. (1998). Using transaction log analysis to improve opac retrieval results. *College & Research Libraries,* 59(1), 39 - 50.

Borgman, C.L., Hirsh, S.G., & Hiller, J. (1996). Rethinking online monitoring methods for information retrieval systems: From search product to search process. *Journal of the American Society for Information Science,* 47(7), 568-583.

Cacheda, F., & Viña, Á. (2001, July). *In Experiences retrieving information in the World Wide Web* (pp. 72-79). Paper presented at the 6th IEEE Symposium on Computers and Communications, Hammamet, Tunisia. IEEE.

Chamberlain, K. (1995, 6 November). *What is grounded theory?* Retrieved 17 September, 2005, from http://kerlins.net/bobbi/research/qualresearch/bibliography/gt.html

Chau, M., Fang, X., & Sheng, O.R.L. (2006). Analyzing the query logs of a Website search engine. *Journal of the American Society for Information Science and Technology,* 56(13), 1363-1376.

Choo, C., Betlor, B., & Turnbull, D. (1998). In A *behavioral model of information seeking on the Web: Preliminary results of a study of how managers and it specialists use the Web* (pp. 290-302). Paper presented at the 61st Annual Meeting of the American Society for Information Science, Pittsburgh, PA. ASIS.

Choo, C., & Turnbull, D. (2000). Information seeking on the Web: An integrated model of browsing and searching. *First Monday,* 5(2), http://firstmonday.org/issues/issue5_2/choo/index.html.

Cooper, M.D. (1998). Design considerations in instrumenting and monitoring Web-based information retrieval systems. *Journal of the American Society for Information Science,* 49(10), 903–919.

Croft, W., Cook, R., & Wilder, D. (1995, 11 - 13 June). In *Providing government information on the internet: Experiences with thomas* (pp. 19-24). Paper presented at the the Digital Libraries Conference, Austin, TX.

Drott, M.C. (1998). *In Using Web server logs to improve site design* (pp. 43 - 50). Paper presented at the the 16th Annual International Conference on Computer Documentation, Quebec, Canada. ACM.

Dumais, S.T. (2002, 7-11 May). *Web experiments and test collections.* Retrieved 20 April, 2003, from http://www2002.org/presentations/dumais.pdf

Efthimiadis, E.N., & Robertson, S.E. (1989). Feedback and interaction in information retrieval. In C. Oppenheim (Ed.), *Perspectives in information management* (pp. 257-272). London: Butterworths.

Fourie, I. (2002, 24 -25 October). *A review of Web information-seeking/searching studies (2000 - 2002): Implications for research in the south* *african context* (pp. 49-75). Paper presented at the Progress in Library and Information Science in Southern Africa: 2d Biannial DISSAnet Conference, Pretoria, South Africa. SAOUG.

Fourie, I., & van den Berg, H. (2003, June). *A story told by nexus transaction logs: What to make of it* (pp. 1 - 19). Paper presented at the 7th Southern African Online Meeting, Muldersdrift, South Africa. SAOUG.

Glaser, B., & Strauss, A. (1967). *The discovery of grounded theory: Strategies for qualitative research*. Chicago, IL: Aldine Publishing Co.

Griffiths, J.R., Hartley, R.J., & Willson, J.P. (2002). An improved method of studying user-system interaction by combining transaction log analysis and protocol analysis. *Information Research*, 7(4), http://InformationR.net/ir/7-4/paper139.html.

Hancock-Beaulieu, M. (2000). Interaction in information searching and retrieval. *Journal of Documentation*, 56(4), 431-439.

Hancock-Beaulieu, M., Robertson, S., & Nielsen, C. (1990). *Evaluation of online catalogues: An assessment of methods* (bl research paper 78). London: The British Library Research and Development Department.

Hargittai, E. (2002). Beyond logs and surveys: In-depth measures of people's web use skills. *Journal of the American Society for Information Science and Technology,* 53(14), 1239-1244.

He, D., Göker, A., & Harper, D.J. (2002). Combining evidence for automatic Web session identification. *Information Processing & Management,* 38(5), 727-742.

Hilbert, D., & Redmiles, D. (1998, 10-13 May). *Agents for collecting application usage data over the Internet* (pp. 149-156). Paper presented at the Second International Conference on Autonomous Agents (Agents '98), Minneapolis/St. Paul, MN.

Hilbert, D., & Redmiles, D. (2001, 9-13 July). *Large-scale collection of usage data to inform design* (pp. 569-576). Paper presented at the Eight IFIP TC 13 Conference on Human-Computer Interaction (INTERACT 2001), Tokyo, Japan.

Jansen, B.J. (2006). Search log analysis: What is it; what's been done; how to do it. *Library and Information Science Research,* 28(3), 407-432.

Jansen, B.J., & Pooch, U. (2001). Web user studies: A review and framework for future work. *Journal of the American Society of Information Science and Technology,* 52(3), 235-246.

Jansen, B.J., Ramadoss, R., Zhang, M., & Zang, N. (2006). Wrapper: An application for evaluating exploratory searching outside of the lab, SIGIR 2006 Workshop on Evaluating Exploratory Search Systems. *The 29th Annual International ACM SIGIR Conference on Research & Development on Information Retrieval (SIGIR2006).* Seattle, Washington, USA.

Jansen, B.J., & Spink, A. (2003, 23 - 26 June). *In An analysis of Web information seeking and use: Documents retrieved versus documents viewed* (pp. 65-69). Paper presented at the 4th International Conference on Internet Computing, Las Vegas, Nevada.

Jansen, B.J., & Spink, A. (2005). How are we searching the World Wide Web? A comparison of nine search engine transaction logs. *Information Processing & Management,* 42(1), 248-263.

Jansen, B.J., Spink, A., & Pedersen, J. (2005). Trend analysis of AltaVista Web searching. *Journal of the American Society for Information Science and Technology,* 56(6), 559-570.

Jansen, B.J., Spink, A., & Saracevic, T. (2000). Real life, real users, and real needs: A study and analysis of user queries on the Web. *Information Processing & Management,* 36(2), 207-227.

Jones, S., Cunningham, S., & McNab, R. (1998, June 1998). *In Usage analysis of a digital library* (pp. 293-294). Paper presented at the the Third ACM Conference on Digital Libraries, Pittsburgh, PA. ACM.

Kelly, D. (2004). *Understanding implicit feedback and document preference: A naturalistic user study*. New Brunswick: Rutgers, The State University of New Jersey.

Kinsella, J., & Bryant, P. (1987). Online public access catalogue research in the united kingdom: An overview. *Library Trends,* 35(4), 619 - 629.

Korfhage, R. (1997). *Information storage and retrieval*. New York: Wiley.

Kurth, M. (1993). The limits and limitations of transaction log analysis. *Library Hi Tech,* 11(2), 98-104.

Lin, S.-J. (2002, 9-11 August). *In Design space of personalized indexing: Enhancing successive Web searching for transmuting information problems* (pp. 1092 - 1100). Paper presented at the Eighth Americas Conference on Information Systems, Dallas, Texas. AIS.

McGrath, J.E. (1994). Methodology matters: Doing research in the behavioral and social sciences. In R. Baecker & W.A.S. Buxton (Eds.), *Readings in human-computer interaction: An interdisciplinary approach* (2nd ed., pp. 152-169). San Mateo, CA: Morgan Kaufman Publishers.

Meister, D., & Sullivan, D. (1967). E*valuation of user reactions to a prototype on-line information retrieval system: Report to nasa by the bunker-ramo corporation*. Report number nasa cr-918. Oak Brook, IL: Bunker-Ramo Corporation.

Millsap, L., & Ferl, T. (1993). Search patterns of remote users: An analysis of opac transaction logs. *Information Technology and Libraries,* 11(3), 321-343.

Miwa, M. (2001, 2-4 February). *In User situations and multiple levels of users goals in information problem solving processes of askeric users* (Vol. 38, pp. 355-371). Paper presented at the the 2001 Annual Meeting of the American Society for Information Sciences and Technology, San Francisco, CA, USA. ASIS.

Montgomery, A., & Faloutsos, C. (2001). Identifying web browsing trends and patterns. *IEEE Computer,* 34(7), 94-95.

Moukdad, H., & Large, A. (2001). Users' perceptions of the web as revealed by transaction log analysis. *Online Information Review,* 25(6), 349-358.

Nicholas, D., Huntington, P., Lievesley, N., & Withey, R. (1999). Cracking the code: Web log analysis. *Online and CD ROM Review,* 23(5), 263-269.

Özmutlu, S., Özmutlu, H.C., & Spink, A. (2003, 23 - 26 June). *In A study of multitasking Web searching* (pp. 145-150). Paper presented at the the IEEE ITCC'03: international Conference on information Technology: Coding and Computing, Las Vegas, Nevada. IEEE.

Park, S., Bae, H., & Lee, J. (2005). End user searching: A Web log analysis of NAVER, a Korean Web search engine. *Library & Information Science Research,* 27(2), 203-221.

Penniman, W.D. (1975, 26-30 October). In C.W.H.L. Tighe (Ed.), *A stochastic process analysis of online user behavior* (pp. 147-148). Paper presented at the The Annual Meeting of the American Society for Information Science, Washington, DC. ASIS.

Peters, T. (1993). The history and development of transaction log analysis. *Library Hi Tech,* 42(11), 41-66.

Phippen, A., Sheppard, L., & Furnell, S. (2004). A practical evaluation of Web analytics. *Internet Research: Electronic Networking Applications and Policy,* 14(4), 284-293.

Pitkow, J.E. (1997, 7-11 April). *In search of reliable usage data on the www* (pp. 1343-1355). Paper presented at the Santa Clara, CA, the Sixth International World Wide Web Conference. Elsevier.

Rice, R.E., & Borgman, C.L. (1983). The use of computer-monitored data in information science. *Journal of the Amercian Society for Information Science,* 44(1), 247-256.

Ross, N., & Wolfram, D. (2000). End user searching on the internet: An analysis of term pair topics submitted to the excite search engine. *Journal of the American Society for Information Science,* 51(10), 949-958.

Sandore, B., Flaherty, P., & Kaske, N.K. (1993). A manifesto regarding the future of transaction log analysis. *Library Hi Tech,* 11(2), 105-111.

Saracevic, T. (1997, 1-6 November). *In Extension and application of the stratified model of information retrieval interaction* (Vol. 34, pp. 313-327). Paper presented at the The Annual Meeting of the American Society for Information Science, Washington, DC.

Silverstein, C., Henzinger, M., Marais, H., & Moricz, M. (1999). Analysis of a very large Web search engine query log. *SIGIR Forum,* 33(1), 6-12.

Spink, A. (2004). Multitasking information behavior and information task switching: An exploratory study. *Journal of Documentation,* 60(3), 336-345.

Spink, A., & Jansen, B.J. (2004). *Web search: Public searching of the Web.* New York: Kluwer.

Spink, A., Wilson, T., Ellis, D., & Ford, F. (1998, April 1998). Modeling users' successive searches in digital environments. *D-Lib Magazine*.

Sullivan, D. (2001, November 6). *Spiderspotting: When a search engine, robot or crawler visits.* Retrieved 5 August, 2003, from http://www.searchenginewatch.com/webmasters/article.php/2168001

Toms, E.G., Freund, L., & Li, C. (2004). Wiire: The Web interactive information retrieval experimentation system prototype. *Information Processing & Management,* 40(4), 655-675.

Wang, P., Berry, M., & Yang, Y. (2003). Mining longitudinal Web queries: Trends and patterns. *Journal of the American Society for Information Science and Technology,* 54(8), 743-758.

Wu, K.-L., Yu, P.S., & Ballman, A. (1998). Speedtracer: A Web usage mining and analysis tool. *IBM Systems Journal,* 37(1), 89-107.

Yuan, W., & Meadow, C.T. (1999). A study of the use of variables in information retrieval user studies. *Journal of the American Society for Information Science,* 50(2), 140-150.

KEY TERMS

Search Log: An electronic record of interactions that have occurred during a searching episode between a Web search engine and users searching for information on that Web search engine.

Search log Analysis (SLA): The use of data collected in a search log to investigate particular research questions concerning interactions among Web users, the Web search engine, or the Web content during searching episodes.

Interactions: The physical expressions of communication exchanges between the searcher and the system.

Search Log Analysis (SLA) Process: A three stage process of collection, preparation and analysis.

APPENDIX A

SQL Query 00:

```
qry_00_no_dups
SELECT tbl_main.uid, tbl_main.date, tbl_main.search_url, Count(tbl_main.search_url)
    AS CountOfsearch_url, First(tbl_main.thetime) AS FirstOfthetime,
    First(tbl_main.qid) AS FirstOfqid INTO tbl_searching_episodes
FROM tbl_main
GROUP BY tbl_main.uid, tbl_main.date, tbl_main.search_url;
```

SQL Query 01:

```
qry_01_unique_ip_number_of_queries
SELECT tbl_searching_episodes.uid
FROM tbl_searching_episodes
GROUP BY tbl_searching_episodes.uid
HAVING (((Count(tbl_searching_episodes.uid))>=100));
```

SQL Query 02:

```
qry_02_remove_large_sessions
DELETE tbl_searching_episodes.qid, tbl_searching_episodes.uid,
tbl_searching_episodes.thetime, tbl_searching_episodes.search_url,
tbl_searching_episodes.qtot, tbl_searching_episodes.uid
FROM tbl_searching_episodes
WHERE (((tbl_searching_episodes.uid)=”[inset values here]”));
```

SQL Query 03:

```
qry_03_list_of_unique_ips
SELECT tbl_searching_episodes.uid, Count(tbl_searching_episodes.search_url) AS
CountOfsearch_url
FROM tbl_searching_episodes
GROUP BY tbl_searching_episodes.uid
ORDER BY Count(tbl_searching_episodes.search_url) DESC;
```

SQL Query 04:

```
qry_04_average_queries_per_user
SELECT Avg(qry_03_list_of_unique_ips.CountOfsearch_url) AS
AvgOfCountOfsearch_url
FROM qry_03_list_of_unique_ips;
```

SQL Query 05:

qry_05_session_length
SELECT qry_03_list_of_unique_ips.CountOfsearch_url,
Count(qry_03_list_of_unique_ips.CountOfsearch_url) AS CountOfCountOfsearch_url
FROM qry_03_list_of_unique_ips
GROUP BY qry_03_list_of_unique_ips.CountOfsearch_url
ORDER BY Count(qry_03_list_of_unique_ips.CountOfsearch_url) DESC;

SQL Query 06:

qry_06_number_of_result_pages
SELECT tbl_searching_episodes.qtot, Count(tbl_searching_episodes.qtot) AS CountOfqtot
FROM tbl_searching_episodes
GROUP BY tbl_searching_episodes.qtot
ORDER BY tbl_searching_episodes.qtot;

SQL Query 07:

qry_07_average_results_pages
SELECT Avg(tbl_searching_episodes.qtot) AS AvgOfqtot
FROM tbl_searching_episodes;

SQL Query 08:

qry_08_repeat_queries
SELECT tbl_searching_episodes.search_url, Count(tbl_searching_episodes.search_url) AS CountOfsearch_url
FROM tbl_searching_episodes
GROUP BY tbl_searching_episodes.search_url
ORDER BY Count(tbl_searching_episodes.search_url) DESC;

SQL Query 09:

qry_09_boolean_queries
UPDATE tbl_searching_episodes SET tbl_searching_episodes.boolean = True
WHERE (((tbl_searching_episodes.search_url) Like " and *" Or (tbl_searching_episodes.search_url) Like "* or *" Or (tbl_searching_episodes.search_url) Like "* and not *"));*

SQL Query 10:

qry_10_query_operators
```
UPDATE tbl_searching_episodes SET tbl_searching_episodes.operator = True
WHERE (((tbl_searching_episodes.search_url) Like ‘*”*’ Or
(tbl_searching_episodes.search_url) Like “*+*” Or (tbl_searching_episodes.search_url)
Like “*[*]*” Or (tbl_searching_episodes.search_url) Like “*[?]*”));
```

SQL Query 11:

qry_11_sum_total_terms
```
SELECT Sum(tblterms.tfreq) AS SumOftfreq
FROM tblterms;
```

SQL Query 12:

qry_12_average_query_length
```
SELECT Avg(tbl_searching_episodes.qry_length) AS AvgOfqry_length
FROM tbl_searching_episodes;
```

SQL Query 13:

qry_13_cooc
```
SELECT tblterms.term, tblterms.term, tblcooc.tot
FROM tblterms INNER JOIN tblcooc ON (tblterms.termid = tblcooc.cid2) AND
(tblterms.termid = tblcooc.cid1)
ORDER BY tblcooc.tot DESC;
```

SQL Query 14:

qry_14_list_of_query_lengths
```
SELECT tbl_searching_episodes.qry_length, Count(tbl_searching_episodes.qry_length)
AS CountOfqry_length
FROM tbl_searching_episodes
GROUP BY tbl_searching_episodes.qry_length
ORDER BY Count(tbl_searching_episodes.qry_length) DESC;
```

SQL Query 15:

qry_15_term_frequencies
```
SELECT tblterms.tfreq
FROM tblterms
GROUP BY tblterms.tfreq
ORDER BY tblterms.tfreq;
```

SQL Query 16:

qry_16_cooc_total
SELECT Sum(tblcooc.tot) AS SumOftot
FROM tblcooc;