How to Define Searching Sessions on Web Search Engines

Bernard J. Jansen
College of Information Sciences and Technology
The Pennsylvania State University
329F IST Building, University Park PA 16802
jjansen@acm.org

Amanda Spink
Faculty of Information Technology
Queensland University of Technology
Gardens Point Campus, 2 George St, GPO Box 2434
Brisbane QLD 4001 Australia
ah.spink@qut.edu.au

Vinish Kathuria
Search Engineer
Infospace, Inc. – Search & Directory
601 108th Ave NE, Ste 1200
Bellevue, WA 98004 USA
Vinish.Kathuria@infospace.com

Sherry Koshman
School of Information Sciences
University of Pittsburgh
610 IS Building, 135 N. Bellefield Avenue
Pittsburgh PA 15260
skoshman@sis.pitt.edu

ABSTRACT

We investigate three methods for defining a session on Web search engines. We examine 2,465,145 interactions from 534,507 Web searchers. We compare defining sessions using: 1) Internet Protocol address and cookie; 2) Internet Protocol address, cookie, and a temporal limit on intra-session interactions; and 3) Internet Protocol address, cookie, and query reformulation patterns. Research results shows that defining sessions by query reformulation along with Internet Protocol address and cookie, provides the best measure, resulting in an 82% increase in the number of sessions. Regardless of the method, mean session length was fewer than three queries and the mean session duration was less than 30 minutes. Implications are that unique sessions may be a better indicator than the common industry metric of unique visitors for measuring search traffic. Results of this research may lead to tools to better support Web searching.

Categories and Subject Descriptors

H.3.3 [1] Information Search and Retrieval – *Search process*

General Terms

Measurement, Experimentation, Human Factors

Keywords

Web sessions, Web queries, query reformulation, Markov states

1. INTRODUCTION

Detecting query reformulations by a Web searcher during a search episode is an important area of research for designing helpful searching systems, recommender systems, personalization, and targeting content to particular users. One can define a search episode on a Web search engine as a temporal series of interactions among a searcher, a Web system, and the content provided by that system within a specific period. During a Web search episode, the user may take several actions including submitting a query, viewing result pages, clicking on URLs, viewing Web documents, and returning to the Web search engine for query reformulation. However, it is possible that one searching episode will be composed of one or more sessions.

We classify a session from a contextual viewpoint as a series of interactions by the user towards addressing a single information need. As such, the session is the critical level of analysis in determining the success or failure of a Web information system. If the information need of the session is satisfied, then one can say that the system (or perhaps the user – system team) was successful. Additionally, an understanding of the contextual information need of a session is key to the development of tools to support Web searching, especially for more complex information needs such as health, ecommerce, and exploratory searching.

Several researchers have analyzed Web searching sessions with the goal of using the information about users' activities to improve the performance of Web search engines. Methods explored by these researchers include both qualitative (i.e., the use of human judges to manually analyze query patterns on usually small samples) and nondeterministic algorithms, typically using large amounts of training data to predict sessions boundaries.

Shneiderman, Byrd and Croft [25] present recommendations for designing Web search engines interfaces that support the searching session strategies of users. Also, Hansen and Shriver [7] examined navigation data using a session-level analysis to cluster search sessions.

Efforts relying on session-level data have taken a variety of approaches. CiteSeer [19] utilizes an agent paradigm to recommend computer science and computer science-related articles based on a user profile. CiteSeer (http://citeseer.ist.psu.edu/) offers a variety of searching assistance, such as related papers, based on searcher

WEBKDD'06, August 20, 2006, Philadelphia, Pennsylvania, USA.

interactions during the session. Jansen and Pooch [12] developed an application for Web search engines that provided targeted searching assistance based on the user interactions during a session. The researchers noted that there are predictable patterns of when searchers seek and implement assistance from the system [9, 10]. These patterns may designate when the searcher is open to assistance from the system, thereby avoiding task interruptions.

Using transaction logs, Anick [1] studied the interactive query reformulation support of the AltaVista search engine for searchers. The researcher used a control group of AltaVista searchers given no query feedback and a feedback group offered twelve refinement terms along with the search results. There was no significant difference in searching performance between the two groups. Conversely, Belkin, et. al., [3] reported that query expansion assistance from the system may be helpful and improve searching performance.

However, an obstacle with all of these applications relying on searching data is determining "exactly what is the session" in practical terms. That is, what is the set of interactions by the user that relates to a single information need? With traditional IR or library systems, one could usually distinguish one user from another user based on a logon. In the Internet environment, how to determine a session between a searcher and a Web search engines is an open question. The difficulty related to both technical and contextual issues.

2. RELATED STUDIES

On the Web, a technological difficulty of how to define a search session is due in part to the stateless nature of the client-server relationship. Most Web search engines servers have used the IP address of the client machine to identify *unique visitors*. With referral sites, Internet service providers (ISP), dynamic Internet Protocol (IP) addressing, and common user terminals it is not always easy to identify a single user session on a Web search engine. Therefore, a single IP address does not always correspond to a single user. However, this approach is commonly used for marketing purposes and Web site traffic reporting (see for example, Nielsen Netranking and iProspect).

In reaction to the dynamically allocated IP's situations, Web search engines have moved to the use of cookies, along with IP addresses, for user identification. The use of cookies minimizes the session identification problem somewhat, but with common access computers (i.e., computers at libraries, schools, labs, office shops, and manufacturing floors which many people share) along with spyware, and cookie management software, one computer may not correspond to one searcher. Additionally, a single searcher may engage a search engine with multiple information needs simultaneously or in rapid succession [30, 31] during a single searching episode. To consider these multiple information needs together presents significant problems for recommender systems and personalized online content.

Consequently, some search engines also use a temporal boundary along with cookies to help address this problem. The idea being that this temporal boundary helps minimize the common user terminal issue, and it also helps delineate repeat searchers to a Web search engine who have returned but with a new information need. However, this approach does not address the multiple information needs during a single searching episode issue. These methods (*IP address; IP and cookie;* and *IP, cookie, and temporal boundary*) all employ a mechanical definition of a session rather than a conceptual definition that defines a searching session within an information seeking task.

Related to the technical constraints, there has been some research into using the query context to define the session. However, this is difficult to do given that the user is typically not available or unwilling to provide deep contextual elaboration.

He, Göker and Harper [8] used contextual information from a Reuters transaction log and a version of the Dempster–Shafer theory in an attempt to identify search engine session boundaries. Using transaction log IP codes and query context, the researchers determined the average Web user session duration was about 12 minutes. Jansen and Spink [13] reported a mean session length of about 15 minutes but with a sizeable percentage of sessions being less than 5 minutes.

Özmutlu and Cavdur [22] attempted to duplicate the findings of [8], but the researchers reported that there were issues relating to implementation, algorithm parameters, and fitness function. Özmutlu, Çavdur, Spink and Özmutlu [23] and Özmutlu and Cavdur [22] investigated the use of neural networks to automatically identify topic changes in sessions, reporting high percentages (72% - 97%) of correct identifications of topic shifts and topic continuations. Özmutlu, et. al. [23] report that neural networks were effective at topic identification, even if the neural network application was trained with data from another search engine transaction log. This line of research involved the use of sophisticated algorithmic approaches or extensive amounts of training data for topic identification. Whether one could obtain comparable results with simpler approaches was not investigated. In addition, these research studies did not contrast the findings of their approaches with other methods of session identification or reformulation classifications.

This study examines three methods of session identification. For real time identification of sessions, one desires the method that is most relatively straightforward but also as accurate as possible. With a method of low computational costs and high accuracy, one could implement such an algorithm for real session identification in Web searching systems.

We compare the results among these three methods of session identification. We also examine quantitative techniques of identifying query reformulations within sessions. We compare the results from our dataset to results reported in other research.

3. RESEARCH QUESTION

Our research question is: *What are the differences in results when using alternative methods for identification of Web search engines sessions?*

We compare three methods for session identification. The methods we use are (1) IP address and cookie; (2) IP address,

cookie, and a temporal cut-off; and (3) IP address, cookie, and context changes. Although there may be other techniques, these three methods represent the major approaches to session identification. We do not evaluate the sole use of an IP address for session identification, as it is commonly known to be inferior to the use of both IP address and cookie.

For this research, we used data from real traffic from an operational Web search engine, Dogpile.com. We explain our data analysis methods in the following section.

3.1.Research Design

3.1.1. Web Data

Dogpile.com (http://www.Dogpile.com/) is a meta-search engine, owned by Infospace, Inc. Meta-search engines provide a unique service by presenting the combined results provided by the various search engines, which have a low rate of overlap [28]. When a searcher submits a query, Dogpile.com simultaneously submits the query to multiple other Web search engines, collects the results from each, removes duplicates results, and aggregates the remaining results into a combined ranked listing using a proprietary algorithm. Dogpile.com integrates the results of the four leading Web search indices (i.e., Ask, Google, MSN, and Yahoo!) along with other search engines into its search results listing.

Dogpile.com has indexes for searching the *Web*, *Images*, *Audio*, and *Video* content, which searchers can access via tabs off the Dogpile.com interface. Dogpile.com also offers query reformulation assistance with alternate query suggestions listed in an *Are You Looking for?* area of the interface.

3.1.2. Data Collection

We collected the records of searcher – system interactions in a transaction log that represents a portion of the searches executed on Dogpile.com on 6 May 2005. The original general transaction log contained 4,056,374 records, each containing seven fields:

- *User Identification*: a code to identify a particular computer
- *Cookie*: an anonymous cookie automatically assigned by the Dogpile.com server to identify unique users on a particular computer.
- *Time of Day*: measured in hours, minutes, and seconds as recorded by the Dogpile.com server on the date of the interaction.
- *Query Terms*: the terms exactly as entered by the given user.
- *Location*: a code representing the geographic location of the user's computer as denoted by the computer's IP address.
- *Source*: the content collection that the user selects to search (e.g., *Web, Images, Audio, News,* or *Video*), with *Web* being the default.
- *Feedback*: a binary code denoting whether or not the query was generated by the *Are You Looking for?* query reformulation assistance provided by Dogpile.com. (see Figure 1).

We imported the original flat ASCII transaction log file of 4,056,374 records into a relational database. We generated a unique identifier for each record. We used four fields (*Time of Day*, *User Identification*, *Cookie*, and *Query*) to locate the initial query and then recreate the sequential series of actions from a particular user, determined by *User Identification* and *Cookie*. An analysis of the dataset shows that the interactions of Dogpile.com searchers was generally similar to Web searching on other Web search engines [16], so we expect the results to be generalizeable.

3.1.3. Data Preparation

The terminology that we use in this research is similar to that used in other Web transaction log studies [c.f., 12, 24] for directed searching on Web search engines.

- *Term:* a series of characters within a query separated by white space or other separator.
- *Query:* string of terms submitted by a searcher in a given instance of interaction with the search engine.
  - *Initial query:* first query submitted in a session by a given user.
  - *Subsequent query:* a query within a session that is not the *initial query*.

At the session level, we deviate from earlier work. In prior studies [c.f., 2, 24], researchers generally defined a *session* as a *series of queries submitted by a user during one episode of interaction between the user and the Web search engine*. Researchers have added certain operational constraints to this definition including whether or not to include the viewing of Web pages [7] and temporal cut-offs between query submissions [26]. Each of these constraints, or lack of constraints, affects what is a *session.* We investigate the effect of some of these constraints in this paper.

How to constrain a session affects other metrics concerning sessions, namely:

- *Session Length*: the number of queries submitted by a searcher during a defined period of interaction with the search engine.

  How one defines the session boundaries is critically important in determining session length.
- *Sessions Duration*: the period from the submission of the *initial query* through the submission of *final query.*

Determining the *initial query* is relatively straightforward. Determining the *final query* again depends on how one defines the session boundaries conditions. For example, if one uses only IP address with no other conditions, than the session duration is the period from the *initial query* until the searcher departed the search engine for the last time (i.e., does not return to the search engine). If one includes other constraints, than there may be multiple sessions by a single searcher within a given episode. However, all are constrained by the recording ability of the Web server.

Unless one has client-side data, search engine logs can only measure the total user time on the search engine, defined as the time spent viewing the first and subsequent results lists and documents, except the final Web document regardless of any other constraints on the session. This final viewing time is not available since the search engine servers record the time stamp. Naturally, the time between visits from the Web document to the server may not have been entirely spent viewing the Web document or interacting with the search engine.

This view of directed search on the Web certainly ignores browsing for information, which one could also include within the session. Bodoff [5] defines b*rowsing* as "actively looking through information (active) or keeping one's eyes open for information (passive), without a particular problem to solve or question to answer (unfocused need)" [p. 70]. Bodoff [5] also provides a nice review of browsing definitions within certain contents and contrasts browsing with directed search, such as that on a Web search engine. However, our focus in this research is on directed searching.

3.2. Data Analysis

3.2.1. Session Analysis Using Multiple Methods

Returning to our research question (*What are the differences in results when using alternative methods for identification of Web search engines sessions?),* we investigated defining sessions using three approaches.

Method 1: IP and Cookie

For the first approach, we defined the session as the period from the first interaction by the searcher with Dogpile.com thorough the last interaction as recorded in the transaction log. We used the searcher's IP address and the browser cookie to determine the *initial query* and all *subsequent queries* to establish *session length*. The *session duration* was the period from the time of the *initial query* to time of the last interaction with the search engine. A change in either IP address or cookie always identified a new session. The algorithm for method 1 is shown in Figure 1.

Figure 1. Method 1 for Identifying sessions.

Algorithm: *IP and Cookie Session Identification*
Assumptions:
1. Null queries and page request queries are removed.
2. Transaction log is sorted by IP address, cookie, and time (ascending order by time).
Input: Record R_i with IP address (IP_i) and cookie (K_i), and record R_{i+1} with IP address (IP_{i+1}) and cookie (K_{i+1}).
Variables: S_x = count of sessions

Output: Session Identification, $\mathbf{S_x}$
begin
Move to R_i

Store values for IP_i, *and* K_i,

$\mathbf{S_x} = 1$

While not end of file
Move to R_{i+1}

If ($IP_i = IP_{i+1}$ *and* $K_i = K_{i+1}$) *then* $\mathbf{S_x}$

Elseif
$\mathbf{S_x} = \mathbf{S_x} + 1$

(R_{i+1} *now becomes* R_i)

Store values for R_{i+1} *as* IP_i, and K_i
end loop
end

Method 2: IP, Cookie, and Temporal Cut-off

For the second approach to session identification, we again used the searcher's IP address and the browser cookie to determine the *initial query* and *subsequent queries*. However, in this method, we used a 30-minute period between interactions as session boundary. For example, if a searcher submitted two queries within a 30 minute period, this searching episode would be one counted as one session. However, if a searcher submitted two queries and the interaction period between each query was longer than 30 minutes, this episode would be counted as two sessions.

We selected the 30-minute period based on the industry standard view of a session (e.g., see OneClick.com and Nielsen Netranking). This 30-minute norm is most likely based on Catledge and Pitkow's reporting that the typical Web session duration was 25.5 minutes on average [6], although this session metric included browsing activities. However, other temporal metrics have been used. Silverstein, Henzinger, Marais and Moricz [26] assigned a temporal cut-off of 5 minutes between interactions as the maximum session duration. Montgomery and Faloutsos [20] used a 125 minute session period, stating that various temporal cut-offs did not substantially affect results. Additionally, Jansen and Spink [13] and He, Göker, and Harper [8] report that the average search engine is about 15 minute based on IP address alone. The algorithm for method 2 is shown in Figure 2.

Figure 2. Method 2 for Identifying sessions.

Algorithm: *IP, Cookie, and Time Identification*
Assumptions:
1. Null queries and page request queries are removed.
2. Transaction log is sorted by IP address, cookie, and time (ascending order by time).
Input: Record R_i with IP address (IP_i), cookie (K_i), and time T_i, and record R_{i+1} with IP address (IP_i), cookie (K_i), and time T_i.

Variables:
D = serial time for 30 minutes
S_x = count of sessions

Output: Search pattern, *SP*

begin
Move to R_i

Store values for IP_i, K_i, *and* T_i

$\mathbf{S_x}$ = 1

While not end of file

Move to R_{i+1}*If (*$IP_i = IP_{i+1}$ *and* $K_i = K_{i+1}$ *and* $T_{i+1} < T_i + D$*) then* $\mathbf{S_x}$

Elseif

$\mathbf{S_x} = \mathbf{S_x} + 1$

(R_{i+1} *now becomes* R_i*)*

Store values for R_{i+1} *as* IP_i, K_i and T_i
end loop
end

Method 3: IP, Cookie, and Content Change

For the third session identification approach, we used a contextual method to identify sessions. We once again used the searcher's IP address and the browser cookie to determine the *initial query* and *subsequent queries*. However, instead of using a temporal cut-off, we used changes in the content of the user queries.

For this method, we assigned each query into a mutually exclusive group based on an IP address, cookie, query content, use of the feedback feature, and query length. The classifications are:

- *Assistance:* the current query was generated by the searcher's selection of an *Are You Looking For?* query (see Figure 1).
- *Content Change*: the current query is identical but executed on another content collection.
- *Generalization:* the current query is on the same topic as the searcher's previous query, but the searcher is now seeking more general information.
- *New:* the query is on a new topic.
- *Reformulation:* the current query is on the same topic as the searcher's previous query and both queries contain common terms.
- *Specialization*: the current query is on the same topic as the searcher's previous query, but the searcher is now seeking more specific information.

The *initial query* (Q_i) from a unique IP address and cookie always identified a new session. In addition, if a *subsequent query* (Q_{i+1}) by a searcher contained no terms in common with the previous query (Q_i), we also deemed this the start of a new session. Naturally, from an information need perspective, these sessions may be related at some level of abstraction. However, with no terms in common, one can also make the case that the information state of the of the user changed, either based on the results from the Web search engine or from other sources [4]. In addition, from a system perspective, two queries with no terms in common represent different executions to the inverted file index and content collection.

We classified each query using an application that evaluated each record in the database. Building from He, Göker, and Harper [8], the algorithm for the application is shown in Figure 3.

Figure 3. Method 3 for Identifying Sessions.

Algorithm: **Search** *Pattern Identification*
Assumptions:
1. Null queries and page request queries are removed.
2. Transaction log is sorted by IP address, cookie, and time (ascending order by time).

Input: Record R_i with IP address (IP_i), cookies (K_i), query Q_i, feedback F_i, and query QL_i; and record R_{i+1} with IP address (IP_{i+1}), cookies (K_{i+1}), query Q_{i+1}, feedback F_{i+1}, and query QL_{i+1}.
Variables:
$B = \{t|t \in Q_i \wedge t \in Q_{i+1}\}$ // terms in common
$C = \{t|t \in Q_i \wedge t \notin Q_{i+1}\}$ // terms that appear in Q_i only
$D = \{t|t \notin Q_i \wedge t \in Q_{i+1}\}$ // terms that appear in Q_{i+1} only
$E = \{1$ if $QL_i = QL_{i+1}\}$ // queries QL_i and QL_{i+1} are the same length; default is 0.
$G = \{1$ if $QL_i > QL_{i+1}\}$ // query QL_i has more terms than QL_{i+1}; default is 0.
$H = \{1$ if $QL_i < QL_{i+1}\}$ // query QL_i has less terms than QL_{i+1}; default is 0.

Output: Search pattern, *SP*
begin

Move to R_i

Store values for IP_i, K_i, Q_i, *Fi, and* QL_i

SP = New *//default value for first* R_i *in record set*

While not end of file
Move to R_{i+1}
If ($IP_i \neq IP_{i+1}$ *and* $K_i \neq K_{i+1}$*) then SP =* New
Elseif
Calculate values for B, C, D, F, G, and H
If $F_{i+1} = 1$ *then SP =* Assistance
Elseif $(B \neq \emptyset \wedge C \neq \emptyset \wedge D = \emptyset \wedge E = 0 \wedge G = 1 \wedge H = 0)$
then SP = Generalization
Elseif $(B \neq \emptyset \wedge C \neq \emptyset \wedge D \neq \emptyset \wedge E = 0 \wedge G = 1 \wedge H = 0)$
then SP = Generalization with Reformulation
Elseif $(B \neq \emptyset \wedge C = \emptyset \wedge D \neq \emptyset \wedge E = 0 \wedge G = 0 \wedge H = 1)$
then SP = Specialization
Elseif $(B \neq \emptyset \wedge C \neq \emptyset \wedge D \neq \emptyset \wedge E = 0 \wedge G = 0 \wedge H = 1)$
then SP = Specialization with Reformulation
Elseif $(B \neq \emptyset \wedge C \neq \emptyset \wedge D \neq \emptyset \wedge E = 1 \wedge G = 0 \wedge H = 0)$
then SP = Reformulation
Elseif $(B \neq \emptyset \wedge C = \emptyset \wedge D = \emptyset \wedge E = 1 \wedge G = 0 \wedge H = 0)$
then SP = Content Change
Elseif SP = New

(R_{i+1} *now becomes* R_i*)*

Store values for R_{i+1} as IP_i, K_i, Q_i ,F_i, and QL_i
end loop

Move to R_1
$\mathbf{S_x} = 0$

While not end of file

If SP = New Then ($S_x = S_x + 1$)

end loop
end

4. RESULTS

We now discuss our results, relating to our research question, focusing on both session length and session duration.

4.1. Session Lengths

We begin by examining differences in session lengths, displayed in Table 1.

Method 1 is the approach used to define a session in many Web searching studies [c.f., 27]. Table 1 shows that more than 79% of the sessions were three or fewer queries, using method 1. Via method 1, the mean session length was 2.85 queries with a standard deviation of 4.43 queries. The maximum session length was 99 and the minimum was 1 query. This finding is similar to other analyses of Web search engines trends. For example, Spink and Jansen [29] reported short sessions during Web searching sessions. Jansen and Spink [14], in their analysis of European searching, noted a similar inclination for short sessions as measured by number of queries submitted. However, AltaVista users conducted slightly longer sessions [17]. Koshman, et al., [18] found that one in five Vivisimo users entered only two queries during their session. Koshman, et al. [18] used IP and cookie on given days to define sessions.

Method 2 has been used by various researchers [c.f., 20, 21, 24, 26], although most employed various time limits, ranging from 5 to 120 minutes. Using method 2, 97% of the sessions were three or fewer queries, which is an 18 %-point increase over method 1. The mean session length was 2.31 queries (15.4% decrease) with a standard deviation of 3.18 queries. The maximum session length was 99 and the minimum was 1 query, which is no change from method 1.

These results parallel more directly the percentage reported by Silverstein, Henzinger, Marais and Moricz [26] that 95% of queries were three queries or fewer using a 5 minute limit between query submissions. Montgomery and Faloutsos [20] defined a session as less than 120 minutes of inactivity between viewings, although they dealt primarily with browsing activity rather than searching. The researchers report that they tried several cutoff values, but the choice did not substantially alter the findings [21]. Catledge and Pitkow [6] report that mean between each user interface events was 9.3 minutes, and they used a session boundary of 25.5 minutes between events, although it is unclear where this temporal boundary came from. Catledge and Pitkow [6] also include browsing activities in their session activities.

Using method 3, we see from Table 1 that 93% of the sessions were three or fewer queries. By way of method 3, the mean session length was 2.31 queries with a standard deviation of 1.56 queries. The maximum session length was 57 and the minimum was 1 query. Note that the mean session length was the same as for method 2, although the standard deviation was about half.

Table 1. Comparing session lengths.

	Method 1: IP and Cookie		Method 2: IP, Cookie, and 30 min. Time Limit		Method 3: IP, Cookie, and Query Content	
Session Length	**Occurrences**	**Percentage**	**Occurrences**	**Percentage**	**Occurrences**	**Percentage**
1	288,231	53.92%	533,950	81.15%	691,672	71.64%
2	88,875	16.63%	81,224	12.34%	153,056	15.85%
3	47,664	8.92%	24,840	3.78%	58,537	6.06%
4	29,345	5.49%	9,219	1.40%	27,134	2.81%
5	19,655	3.68%	3,822	0.58%	14,168	1.47%
6	13,325	2.49%	1,755	0.27%	7,745	0.80%
7	9,549	1.79%	944	0.14%	4,430	0.46%
8	7,169	1.34%	622	0.09%	2,791	0.29%
9	5,497	1.03%	442	0.07%	1,769	0.18%
10	4,130	0.77%	331	0.05%	1,193	0.12%
> 10	21,067	3.94%	871	0.13%	2,944	0.30%
	534,507	100.00%	658,020	100.00%	965,439	100.00%

Generally, it appears that method 3 provides a more granular definition of the session based on the reduced variance in the number of queries per session. Using 534,507 sessions as the base, method 2 resulted in a 23% increase in the number of sessions, and method 3 resulted an 82% increase in sessions.

We investigated whether these three methods produced significantly different results by performing a chi square test. The chi-square is a non-parametric test of statistical significance. The chi-square test tells us whether or not samples are different enough in some characteristic from which we can generalize that the populations are also different.

A chi-square goodness of fit shows that the three methods are statistically different, ($\chi^2(10) = 29.73$, $p<0.01$; critical value of $\chi^2 = 23.209$). So, the methods are significantly dissimilar in their classification of sessions by number of queries

4.2. Session Durations

What is the effect of these methods on session duration? Examining session durations, we see in Table 2 that method 1 shows a large percentage of very short session durations.

The mean session duration was 26 minutes and 32 seconds, with a standard deviation of 1 hour, 36 minutes and 25 seconds. The maximum session was just under 24 hours (23:57:51), and the minimum session was 0 (i.e., the user submitted one query and performed no other search activity on the search engine during the session). This is more than twice that reported by He et al [8], who reported a session duration of 12 minutes.

Using method 2, the absolute numbers have increased, but the percentages of very short session durations remains relatively stable. However, the mean session duration was 6 minutes and 36 seconds, with a standard deviation of 16 minutes and 5 seconds. This is closer to the large number of sessions at approximately 5 minutes reported by Jansen and Spink [13]. The maximum session was just under 24 hours (23:57:24).

As with method 1, the maximum session length is cause for concern, as it seems highly unlikely that a single searcher would spend 24 hours submitting queries to a search engine. More than likely, these methods are inadvertently combining sessions or the database still contains agent submissions.

Using method 3, the percentages of very short session durations again remains relatively stable. The mean session duration was 5 minutes and 15 seconds, with a standard deviation of 39 minutes and 22 seconds. The maximum session duration was again just under 24 hours (23:41:53).

Comparing the mean session durations, the mean using method 1 is 333% greater than the mean session duration using method 2 and 420% greater than the mean session duration using method 3. This outcome is in contrast to that reported by [20] where changes in temporal cut-offs for the session boundaries did not substantially alter results.

5. DISCUSSION

We explored three alternative methods for detection of session boundaries using 2,465,145 interactions from 534,507 users of Dogpile.com recorded on 6 May 2005. We compared three methods of session identification (1) *using IP address and cookie*, (2) *IP address, cookie, and a temporal limit on intra-session interactions*, and (3) *IP address, cookie, and query reformulation patterns*.

Table 2. Comparing session durations.

	Method 1: IP and Cookie		Method 2: IP, Cookie, and 30 min. Time Limit		Method 3: IP, Cookie, and Query Content	
Session Duration	**Occurrences**	**Percentage**	**Occurrences**	**Percentage**	**Occurrences**	**Percentage**
< 1 minute	302,653	56.62%	372,983	56.68%	794,765	82.32%
1 to < 5 minutes	83,236	15.57%	93,251	14.17%	86,358	8.94%
5 to < 10 minutes	36,347	6.80%	55,956	8.50%	28,044	2.90%
10 to < 15 minutes	19,806	3.71%	36,020	5.47%	12,277	1.27%
15 to < 30 minutes	27,210	5.09%	61,767	9.39%	13,752	1.42%
30 to < 60 minutes	18,441	3.45%	30,790	4.68%	12,628	1.31%
60 to < 120 minutes	14,236	2.66%	6,615	1.01%	7,524	0.78%
120 to < 180 minutes	8,262	1.55%	506	0.08%	3,320	0.34%
180 to < 240 minutes	5,901	1.10%	76	0.01%	1,919	0.20%
> 240 minutes	18,415	3.45%	56	0.01%	4,852	0.50%
	534,507	100.00%	658,020	100.00%	965,439	100.00%

Table 3:Query reformulation.

Search Patterns	Occurrence	%	Occurrence (excluding *New*)	% (excluding *New*)
New	964,780	63.34%	-	-
Reformulation	126,901	8.33%	126,901	22.73%
Assistance	124,195	8.15%	124,195	22.25%
Specialization	90,893	5.97%	90,893	16.28%
Content change	65,949	4.33%	65,949	11.81%
Specialization with reformulation	55,531	3.65%	55,531	9.95%
Generalization with reformulation	54,637	3.59%	54,637	9.78%
Generalization	40,186	2.64%	40,186	7.20%
	1,523,072	100.00%	558,292	100.00%

Our results show that defining sessions by query content (method 3) provides the best session identification with an extremely high accuracy rate. Comparatively, method 1 appears to extend artificially both session length and duration. Method 2 appears to shorten artificially session length and duration. By relying on IP address and cookie as a basis, plus content changes between queries, method 3 provides the best contextual identification of Web sessions within a user episode on a Web search engine.

Method 3, using IP address, cookie, and query content changes, appears to provide the most detailed method for session identification with both session length and session duration. Since the method does not rely on probability methods, it can be calculated in real time with near total accuracy of new session identification. Using this content approach, Web search systems can develop automated assistance interfaces, such as reported in [11] that provide session level searching assistance to Web engine users.

As an example, Table 3 presents the query modification executed by searchers during their searching episodes. We see from Table 3 that more than 8% of the query modifications were for *Reformulation*, with another approximately 8% of query modifications resulting from system *Assistance*. If we exclude the *New* queries, *Reformulation* and *Assistance* account for nearly 45% of all query modifications. This finding would seem to indicate that a substantial portion of searchers go through a process of defining their information need by exploring various terms and system feedback to modify the query as an expression of their information need. Another 16% of query modifications are *Specialization*, supporting prior reports that precision is a primary concern for Web searchers [15]. With this tighter view of a session, Web search engines can more effectively personalize for searching assistance, content, or online advertising.

The detection of Web searching sessions is a critical area of research for developing more supportive searching systems, especially in the more complex searching environments of exploratory searching and multitasking. The method presented in this research relies on the content of searchers' queries, along with other data collected by the search engine, to identify searching sessions. The method is advantageous for real-time system implementation.

6. CONCLUSIONS

For future research, these algorithms may be used as models to facilitate cross-system investigations. An attempt to standardize session detection would also enhance comparative transaction log analyses. We are currently conducting qualitative analysis of Dogpile users' query reformulation that we will compare with the results reported in this paper. Also, several searcher – system interactions can be recorded by the Web search engine server. However, there are other actions, such as Back, Forward, Bookmark, Scrolling, among others, that occur on the client-side computer. The server does not record these actions. We are investigating the development of server-client tools that can monitor the entire set of searcher actions during a session.

ACKNOWLEDGMENTS

We thank Infospace, Inc. for providing the Web search engine transaction log data without which we could not have conducted this research. We also thank Ms. Danielle Booth for coding the algorithm for method 3 presented in this manuscript. Portions of this research funded by the U.S. Department of the Air Force, AFRL, FA9550-60-10328.

7. REFERENCES

[1] Anick, P., Using Terminological Feedback for Web Search Refinement - A Log-Based Study, in *Proceedings of the Twenty-Sixth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2003. Toronto, Canada. 28 July - 1 August. pp. 88-95.

[2] Beitzel, S. M., Jensen, E. C., Chowdhury, A., Grossman, D., and Frieder, O., Hourly analysis of a very large topically categorized web query log, in *Proceedings of the 27th annual international conference on Research and development in information retrieval*, 2004. Sheffield, U.K. 25-29 July. pp. 321 - 328.

[3] Belkin, N., Cool, C., Kelly, D., Lee, H.-J., Muresan, G., Tang, M.-C., and Yuan, X.-J., Query length in interactive information retrieval, in *Proceedings of the 26th Annual International ACM Conference on*

Research and Development in Information Retrieval, 2003. Toronto, Canada. 28 July - 1 August. pp. 205 - 212.

[4] Belkin, N., Oddy, R., and Brooks, H., ASK for Information Retrieval, Parts 1 & 2, *Journal of Documentation*, vol. 38, pp. 61-71, 145-164, 1982.

[5] Bodoff, D., Relevance for Browsing, Relevance for Searching, *Journal of the American Society of Information Science and Technology*, vol. 57, pp. 69–86, 2006.

[6] Catledge, L. D. and Pitkow, J. E., Characterizing browsing strategies in the World Wide Web, *Computer Network and ISDN Systems*, vol. 27, pp. 1065-1073, 1995.

[7] Hansen, M. H. and Shriver, E., Using navigation data to improve IR functions in the context of web search, in *Proceedings of the tenth international conference on Information and Knowledge Management*, 2001. Atlanta, Georgia, USA. October. pp. 135 - 142.

[8] He, D., Göker, A., and Harper, D. J., Combining Evidence for Automatic Web Session Identification, *Information Processing & Management*, vol. 38, pp. 727-742, 2002.

[9] Jansen, B. J., Seeking and implementing automated assistance during the search process, *Information Processing & Management*, vol. 41, pp. 909-928, 2005.

[10] Jansen, B. J., Using temporal patterns of interactions to design effective automated searching assistance systems, *Communications of the ACM*, vol. 49, pp. 72-74, 2006.

[11] 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.

[12] Jansen, B. J. and Pooch, U., Web User Studies: A Review and Framework for Future Work, *Journal of the American Society of Information Science and Technology*, vol. 52, pp. 235-246, 2001.

[13] Jansen, B. J. and Spink, A., An Analysis of Web Information Seeking and Use: Documents Retrieved Versus Documents Viewed, in *Proceedings of 4th International Conference on Internet Computing*, 2003. Las Vegas, Nevada. 23 - 26 June. pp. 65-69.

[14] Jansen, B. J. and Spink, A., An Analysis of Web Searching By European Alltheweb.com Users, *Information Processing & Management*, vol. 41, pp. 361-381, 2005.

[15] Jansen, B. J. and Spink, A., How are we searching the World Wide Web? A comparison of nine search engine transaction logs, *Information Processing & Management*, vol. 42, pp. 248-263, 2005.

[16] Jansen, B. J., Spink, A., Blakely, C., and Koshman, S., Web Searcher Interaction with the Dogpile.com Meta-Search Engine, *Journal of the American Society for Information Science and Technology*, forthcoming.

[17] 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.

[18] Koshman, S., Spink, A., Jansen, B. J., Park, M., and Field, C., Web Searching on the Vivisimo Search Engine, *Journal of the American Society of Information Science and Technology*, forthcoming.

[19] Lawrence, S., Giles, C. L., and Bollacker, K., Digital Libraries and Autonomous Citation Indexing, *IEEE Computer*, vol. 32, pp. 67-71, 1999.

[20] Montgomery, A. and Faloutsos, C., Identifying web browsing trends and patterns, *IEEE Computer*, vol. 34, pp. 94-95, 2001.

[21] Montgomery, A. and Faloutsos, C., *Trends and patterns of WWW browsing behaviour*, Accessed on 6 October 2005 on the World Wide Web at http://pages.cpsc.ucalgary.ca/~saul/personal/other_pubs/web_trends.pdf.

[22] Özmutlu, H. C. and Cavdur, F., Application of automatic topic identification on Excite Web search engine data logs, *Information Processing & Management*, vol. 41, pp. 1243-1262, 2005.

[23] Özmutlu, H. C., Çavdur, F., Spink, A., and Özmutlu, S., Cross Validation of Neural Network Applications for Automatic New Topic Identification, in *Proceedings of the Association for the American Society of Information Science and Technology (ASIST 2005)*, 2005. Charlotte, NC. 31 October - 3 November. pp. 1-10.

[24] 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.

[25] Shneiderman, B., Byrd, D., and Croft, W. B., Sorting out searching: a user-interface framework for text searches, *Communications of the ACM*, vol. 41, pp. 95 - 98, 1998.

[26] 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.

[27] Spink, A. and Jansen, B. J., *Web Search: Public Searching of the Web*. New York: Kluwer, 2004.

[28] Spink, A., Jansen, B. J., Blakely, C., and Koshman, S., A Study of Results Overlap and Uniqueness Among Major Web Search Engines, *Information Processing & Management*, forthcoming.

[29] Spink, A., Jansen, B. J., Wolfram, D., and Saracevic, T., From E-sex to E-commerce: Web Search Changes, *IEEE Computer*, vol. 35, pp. 107-111, 2002.

[30] Spink, A., Özmutlu, H. C., and Özmutlu, S., Multitasking information seeking and searching processes, *Journal of the American Society for Information Science and Technology*, vol. 53, pp. 639-652, 2002.

[31] Spink, A., Park, M., Jansen, B. J., and Pedersen, J., Multitasking during web search sessions, *Information Processing & Management*, vol. 42, pp. 264-275, 2005.