The Impact of Site Structure and User Environment on Session Reconstruction in Web Usage Analysis

Bettina Berendt[1], Bamshad Mobasher[2], Miki Nakagawa[2], Myra Spiliopoulou[3]

[1] Humboldt University Berlin, Inst. of Information Systems, berendt@wiwi.hu-berlin.de
[2] DePaul University, Dept. of Computer Science, mobasher|miki@cs.depaul.edu
[3] Leipzig Graduate School of Management, Dept. of E-Business myra@ebusiness.hhl.de

Abstract

The analysis of user behavior on the Web presupposes a reliable reconstruction of the users' navigational activities. Cookies and server-generated session identifiers have been designed to allow a faithful session reconstruction. However, in the absence of reliable methods, analysts must rely on heuristics methods (a) to identify unique visitors to a site, and (b) to distinguish among the activities of such users during independent sessions. The characteristics of the site, such as the site structure, as well as the methods used for data collection (e.g., the existence of cookies and reliable synchronization across multiple servers) may necessitate the use of different types of heuristics. In this study, we extend our work on the reliability of sessionizing mechanisms, by investigating the impact of site structure on the quality of constructed sessions. Specifically, we juxtapose sessionizing on a frame-based and a frame-free version of a site. We investigate the behavior of cookies, server-generated session identification, and heuristics that exploit session duration, page stay time and page linkage. Different measures of session reconstruction quality, as well as experiments on the impact on the prediction of frequent entry and exit pages, show that different reconstruction heuristics can be recommended depending on the characteristics of the site. We also present first results on the impact of session reconstruction heuristics on predictive applications such as Web personalization.

Keywords: Data preparation, sessionization heuristics, Web usage mining

1 Introduction

The quality of the patterns discovered in data analysis depends on the quality of the data on which mining is performed. In Web usage analysis, these data are the sessions of the site visitors. The reliable reconstruction of the visitors' activities in a Web site involves a correct mapping of activities to different individuals and a correct separation of the activities belonging to different visits of the same individual.

In [BMSW01, SMBN02], we describe how the two aspects of reliable session reconstruction are supported by (a) proactive mechanisms that enforce correct mappings during the activities of each visitor and (b) reactive heuristics that perform the mappings a posteriori. Cookie identifiers and session identifiers generated by Web application servers belong to the first category. The specification of upper thresholds on total visit time or on total page stay time are examples of heuristics of the second category. A collection and discussion of such heuristics can be found in [CMS99]. Section 2.2 of the current paper contains a formalization of these heuristics.

Padmanabhan, Zheng, and Kimbrough stress the importance of correct session reconstruction by comparing different definitions of the notion of "session" and comparing their predictive accuracy [PZK01, ZPK02]. However, they do not elaborate on the performance of different mechanisms for session reconstruction.

In [BMSW01], we have proposed a set of measures for the comparison of sessions generated by different heuristics (see section 3). In [BMSW01, SMBN02], we used these measures to compare the results of

proactive and reactive heuristics on a frame-based site. Our experiments showed that proactive mechanisms (like cookies) allow for a much more reliable session reconstruction than reactive ones.

Increasingly, commercial and (to a lesser degree) non-commercial sites have been relying on cookies as a mechanism for the identification of unique users. However, there is still a large proportion of sites who do not use such proactive mechanisms. This is in part due to regulations and self-regulations on e-privacy, particularly in Europe, and in part due to a lack of adequate infrastructural support in data collection and analysis. Furthermore, cookies, by themselves, are not adequate for the correct splitting of a visitor's activities into sessions, and even a fewer proportion of sites currently use server-supported proactive mechanisms for sessionization. As long as no proactive, privacy-conformant mechanism has become commonplace among Web servers, reactive heuristics will therefore remain essential for reliable session reconstruction for usage analysis.

In our previous work [SMBN02], we have observed that the framesets of a frame-based site have a serious impact on the mapping of the activities of each user to distinct sessions. Hence, in this study, we investigate the performance of the sessionizing heuristics for a frame-based and a frame-free version of a site (section 4). Section 5 investigates the impact of the user environment, more specifically the relation between cookies and IP address and user agent.

Since session reconstruction is a preparatory step for data analysis, the "performance" of the heuristics should refer to the predictive power of the sessions they generate. Section 6 describes experiments on the effects on the data mining tasks of (a) predicting exit pages and of (b) recommending a page based on these profiles. The last section concludes our study.

2 Heuristic methods for session reconstruction

Heuristic methods for session reconstruction must fulfill two tasks: First, all activities performed by the same physical person should be grouped together. Second, all activities belonging to the same visit should be placed into the same group. Knowledge about a user's identity is not necessary to fulfill these tasks. However, a mechanism for distinguishing among different users is indeed needed.

In accordance with W3C (1999), we term as *(server) session* or *visit* the group of activities performed by a user from the moment she enters the site to the moment she leaves it. Since a user may visit a site more than once, the Web server log records multiple sessions for each user. We use the name *user activity log* for the sequence of logged activities belonging to the same user. Thus, *sessionizing* is the process of segmenting the user activity log of each user into sessions. A *sessionization heuristic* is a method for performing such a segmentation on the basis of assumption about users' behavior or the site characteristics.

The goal of a heuristic is the faithful reconstruction of the *real sessions*, where a real session is the sequence of activities performed by one user during one visit at the site. We denote the dataset of real sessions as $\mathcal{R}$. A sessionization heuristic h attempts to assign activities to users and to identify the ends of each user visit, i.e. to partition sequences of activities of the same user into sessions. The result is a dataset of *constructed sessions*, which we denote as $\mathcal{C} \equiv \mathcal{C}_h$. For the ideal heuristic, $\mathcal{C} \equiv \mathcal{C}_h = \mathcal{R}$.

2.1 Mapping activities to users and segmenting into sessions

The analysis of Web usage does not require knowledge about a user's identity. However, it is necessary to distinguish among different users. The information available according to the HTTP standard is not adequate to distinguish among users from the same host, proxy, or anonymizer. The most widespread remedy amounts to the usage of cookies. A persistent cookie is a unique identifier assigned by the Web server to each client agent accessing the site (or other sites associated with the same domain) for the first time. This identifier is stored on the client side and transmitted back to the server upon subsequent visits to the server by the same client. A cookie is a *proactive data preparation strategy* because the assignment of a user identification to requests is taken care of *while* the user accesses the site. However, while a cookie provides for user identification across multiple visits to the site, it does not mark these visits' boundaries.

The proactive approach to session identification may also involve the use of *embedded session IDs*. Such session IDs are implemented as an extension of the Web server which assigns a unique identifier to each active

client process accessing the server. This identifier is attached to each request made by the user's client to the server (e.g., by URL rewriting), thus allowing for the unique assignment of requests to users during one visit. The identifier expires when the user's client process is terminated, when the connection is broken or when a timeout occurs. Its expiration determines the end of the session. Other proactive strategies include user authentication / registration and client agent data collection.

In contrast to proactive strategies, *reactive strategies* reconstruct the assignment of a user (or session) identification to requests *after* the requests have been recorded, based on the "user environment" information recorded in the Web server's log.

The most widespread log formats for HTTP servers are the W3C common and extended log file formats. In the common log file format (cf. http://www.w3.org/Daemon/User/Config/Logging.html), the only recorded data related to a user as a person are the IP address or DNS hostname of the user's host or proxy. The extended log file format (see http://iishelp.web.cern.ch/IISHelp/iis/htm/core/iiintlg.htm) also allows the recording of the user's software agent (browser or batch client) that performs the requests on her behalf, as well as the "referrer" URL, i.e. the page from which a request was initiated. However, partitioning by IP+agent is not guaranteed to perform user identification correctly, since several users may be accessing the server from the same IP+agent. Also, it cannot recognize session boundaries.

Therefore, a second step is required that generates a further partitioning into (constructed) sessions. Since this second step is needed for logs partitioned by cookies as well as for logs partitioned by IP+agent, the same *sessionization heuristics* can be employed. These too are reactive strategies.

2.2 A selection of sessionization heuristics

In the present study, we evaluate the performance of the following heuristics:

h1: Time-oriented heuristic: The duration of a session may not exceed a threshold θ.
This heuristic has its origins in research on the mean inactivity time within a site [CP95].

h2: Time-oriented heuristic: The time spent on a page may not exceed a threshold δ.
Heuristics of this type are used in [CMS99, SF99].

href: Referrer-based heuristic: Let p and q be two consecutive page requests, with p belonging to a session S. Let t_p and t_q denote the timestamps for p and q, respectively. Then, q will be added to S if the referrer for q was previously invoked within S, or if the referrer is undefined and $(t_q - t_p) \leq \Delta$, for a specified time delay Δ. Otherwise, q is added to a new constructed session.

The "undefined" referrer ("-" in the log) is usually introduced by a server-dependent process. In many logs, this may be recorded in various situations: (1) As the referrer of the start page, or of a page that was entered after a brief excursion to a sub-site or a foreign server. This may happen, for example, because a site does not record external referrers. (2) As the referrer of a typed-in or bookmarked URL. (3) When a frameset page is reloaded in mid session. (4) For all these pages, when they are reached via the back button during the real session. (5) In a frame-based site: as the referrer of the first frames that are loaded when the start page containing the top frameset is requested. (6) The access to a page was invoked by certain external processes such as a hyperlink from an email client or from within a non-HTML document.

The time delay Δ in the above definition is necessary to allow for proper loading of frameset pages whose referrer is undefined, and to account for other situations resulting in mid-session requests with undefined referrers.

In [SMBN02], we considered the first two heuristics and an earlier version of **href**, comparing their performance on data with prior cookie-based user identification, to their performance on data with prior IP+agent partitioning. The comparison used data from a site that employed frames.

In the present study, we have considered the three heuristics in the two settings *frame-based* and *frame-free*, by comparing the results of the previous setting with cookie identifiers in a site with frames with cookie-identified data from the same site in a frame-free version. In both settings, we have used the standard values $\theta = 30$ minutes maximum total duration for **h1**, and $\delta = 10$ minutes maximum page stay time for

h2 (see [CP95, CMS99, SF99]). Previous experiments indicate a high robustness of these heuristics with respect to variations in the threshold parameters, and superior performance for the two values chosen. In both settings, we have used a default value of 10 seconds for **href**'s Δ, and also investigated the effect of varying this value.

3 Measures of session reconstruction quality

Intuitively, the perfect heuristic would reconstruct all sessions by placing all activities of each user during each visit—and only these—into the same session. Reactive heuristics do not have adequate information for such a perfect assignment of activities to sessions. Thus, it is necessary to quantify the performance of each heuristic with respect to the quality of *all* sessions it builds.

The measures we use quantify the successful mappings of real sessions to constructed sessions, i.e. the "reconstructions of real sessions". In particular, a measure M evaluates a heuristic h based on the difference between $\mathcal{C}_h$ and $\mathcal{R}$. It assigns to h a value $M(h) \in [0,1]$ so that the score for the perfect heuristic ph is $M(ph) = 1$.

In [BMSW01, SMBN02], we have proposed four "categorical" measures that reflect the number of real sessions that are reconstructed by a heuristic *in their entirety*, and two "gradual" measures that take account of the extent to which the real sessions are reconstructed. Here, we reinterpret the categorical measures as recall measures, and extend the framework by the corresponding precision measures. This allows a more differentiated analysis of reconstruction quality.

Categorical measures. Categorical measures enumerate the real sessions that were recognized by the heuristics as distinct visits and were thus mapped *into* constructed sessions, i.e. they are contained in some constructed session.

- The **complete reconstruction** measure $M_{cr}(h)$ returns the number of real sessions contained in some constructed session, divided by the total number of real sessions. A session r is contained in a session c if and only if all its elements are in c, in their correct order, with no intervening foreign elements.

This measure is not very specific, since any number of real sessions may be contained in one constructed session without affecting its value. Therefore, further elaboration is necessary. We consider two types of refinement: On the one hand, we design more restrictive measures, in which the entry or exit page of the real session has been identified as such in the constructed session. On the other hand, we juxtapose the number of real sessions thus considered against either all real sessions or all constructed sessions.

The first type of refinement reflects the fact that the correct identification of the entry or the exit page is essential for many applications. If both the entry and the exit page of a real session are guessed correctly, then the corresponding constructed session is identical to the real one.

The second type of refinement corresponds to the notions of recall and precision. Our algorithms ensure that if a reconstructed session contains a real session and has the same entry (exit) page, then there is only one such constructed session. Therefore, this can be interpreted to mean that the constructed session is the (unique) "correct guess" of that real session. So the corresponding measures can be interpreted as *recall* measures: the number of correct guesses divided by the total number of correct items $|\mathcal{R}|$. We complement this by the corresponding *precision* measures: the number of correct guesses divided by the total number of guesses $|\mathcal{C}_h|$.

We therefore define:

- **complete reconstruction with correct entry page – recall**:
  $M^{recall}_{cr,entry}(h)$ is the number of real sessions mapped into a constructed session with the same entry page as the real session, divided by the number of real sessions.

- **complete reconstruction with correct exit page – recall**:
  $M^{recall}_{cr,exit}(h)$ is the number of real sessions mapped into a constructed session with the same exit page as the real session, divided by the number of real sessions.

- **identical reconstruction – recall**:
  $M^{recall}_{cr,entry-exit}(h)$ is the number of real sessions that appear in $\mathcal{C}_h$, i.e. the intersection of $\mathcal{R}$ and $\mathcal{C}_h$, divided by the number of real sessions.

- **complete reconstruction with correct entry page – precision**:
  $M^{precision}_{cr,entry}(h)$ is the number of constructed sessions that contain a real session and have the same entry page with it, divided by the number of constructed sessions.

- **complete reconstruction with correct exit page – precision**:
  $M^{precision}_{cr,exit}(h)$ is the number of constructed sessions that contain a real session and have the same exit page with it, divided by the number of constructed sessions.

- **identical reconstruction – precision**:
  $M^{precision}_{cr,entry-exit}(h)$ is the number of sessions in the intersection of $\mathcal{R}$ and $\mathcal{C}_h$, divided by the number of constructed sessions.

Categorical measures are relevant for applications in which the faithful reconstruction of the real sessions in their entirety is necessary for the analysis. This includes the evaluation of user satisfaction with a site, and the analysis of the improvement potential of the site as a whole.

Gradual measures. For some application domains, categorical measures are too restrictive. For example, consider the page prefetching problem, in which the next request must be predicted given the requests performed thus far. This only requires the correct reconstruction of previous requests in the same session, usually without recourse to their exact sequence. Similarly, for market basket analysis or recommender systems, the identification of as many co-occurrences of pages as possible is more relevant than the correct identification of the exact sequence of actions, or the correct identification of entry and exit pages. For such applications, we use *gradual measures*.

Gradual measures are based on the overlap between real and reconstructed sessions. For each real session, we find the constructed session that has the maximum number of common elements with it. We normalize this value by dividing it by the length of the real session. Then, we define the **average maximal degree of overlap** $M_o(h)$ as the average of these overlap values over all real sessions.

The measure $M_o(h)$ does not take the number of dissimilar elements between real and constructed session into account. Hence, a heuristic that produces only one constructed session will acquire the highest score 1 for $M_o(h)$. To alleviate this problem, we consider the similarity between real and constructed sessions: For each real session, we again find the constructed session that has the maximum number of common elements with it, but we divide it by the number of elements in the union of real and constructed session. The **average maximal degree of similarity** is the average of these values over all real sessions.

4 Impact of site structure on session reconstruction quality

We first compared the set of reconstructed sessions produced by each heuristic to the "real" sessions. In the analyzed site, a cookie-based mechanism was used for user identification, and session IDs for splitting a users activities into sessions. This combination defined the reference set $\mathcal{R}$ of real sessions. The session identifiers were then removed, and each of the heuristics h described in section 2.2 was employed to produce a set of constructed sessions $\mathcal{C}_h$. By juxtaposing the reconstruction results for a frame-based and a frame-free version of a site, we show that the frame-free version allow for more reliable session reconstruction. At the same time, different heuristics are affected by framesets to a different extent.

The two datasets describe the usage of the same university site, once in a frame-based, and once in a frame-free version. They contained 174660 and 115434 requests, respectively.

The preprocessing of the data included removal of navigations by known robots as well as well-behaved robots (those accessing the `robots.txt` page). We expect that the remaining robots constitute a negligible percentage of sessions. Between 1 and 2% of all users rejected cookies; their requests were removed.

		$\mathcal{C}^{fb}$				$\mathcal{C}^{ff}$		
	$\mathcal{R}^{fb}$	**h1**	**h2**	**href**	$\mathcal{R}^{ff}$	**h1**	**h2**	**href**
Number of sessions	13829	14234	15971	41117	20950	20050	22149	27707
Average session duration	31:56	7:05	3:49	7:12	21:31	5:24	2:55	7:49
Median session duration – both (min:sec) –	1:39	2:59	1:35	0:30	1:14	2:09	1:16	0:48
Average page stay time	2:12	0:44	0:18	1:20	4:59	1:45	0:47	2:00
Median page stay time – both (min:sec) –	0:12	0:16	0:11	0:07	0:21	0:28	0:21	0:16
Average session length	12.63	12.18	10.85	4.21	5.51	5.76	5.21	4.17
Median session length – both (no. of pages) –	7	8	7	2	3	4	3	2

Table 1: Base statistics for the sessions in the frame-based (*fb*) and in the frame-free (*ff*) site version

4.1 Session Statistics for a Frame-based and a Frame-free Site

Table 1 shows the basic statistics for the two site versions. We report median values because earlier experiments [BMSW01, SMBN02] showed that averages were not representative of the distribution of values in the frame-based dataset. The discrepancy between median and average values holds both for the frame-based and the frame-free version and is most remarkable for session duration. It is apparent that a heuristic making a good approximation of the median values of the real sessions captures the characteristics of the real sessions much better than one that only approximates the average values.

With respect to the approximation of average values, the three heuristics behave similarly for the frame-based and the frame-free version of the site. The value distributions they produce are smoother than the distribution of the real values in the sense that the medians of the constructed sessions are closer to the corresponding averages than is the version for the real sessions. In particular: (a) All heuristics underestimate the average session duration in both the frame-based and the frame-free version. (b) All heuristics underestimate the average page stay time for both site versions. (c) **h1** and **h2** return better approximations of average session length than **href**, with **h1** returning the best approximations.

With respect to the approximation of median values, the three heuristics behave similarly for the frame-based and the frame-free version of the site. In particular: (a) Heuristic **h2** makes the best approximation of all median values in both the frame-based and the frame-free version. (b) Heuristic **h1** overestimates all median values, while **href** underestimates them.

The heuristic **href** makes better approximations of the medians in the frame-free than in the frame-based version of the site. This was expected; pages in the frame-free site have fewer objects than those in the frame-based site, so that **href** encounters fewer misleading referrers.

With respect to the approximation of the number of real sessions, Heuristic **h1** provides the best approximation of the number of real sessions. Heuristic **h2** overestimates this number for both the frame-based and the frame-free version. Overestimation is a more severe problem for **href**: It produces more than three times as many sessions as there actually are. Its performance is much better the frame-free version, where "only" 25% more sessions are produced.

To explain the performance of **href**, we cross-compare the disproportionately large number of sessions, the very low median values for session duration, page stay and session length in pages, and the low average session length for this heuristic. Those numbers indicate that **href** produces a very large number of tiny reconstructed sessions by splitting real sessions. In the frame-based version of the site, the median length of real sessions is 7, and the corresponding median for the sessions produced by **href** is 2, implying that each session in $\mathcal{R}^{fb}$ corresponds to around 3 sessions in $\mathcal{C}^{fb}_{href}$. In the frame-free version, the median length of real sessions is only 3, so that the tiny sessions produced by **href** produce better approximations for the statistics upon the real sessions.

In terms of applicability of the heuristics, the basic statistics indicate that the referrer heuristic should

not be used in a frame-based version. The two time-based heuristics are less affected by the presence of framesets and can be used in both types of site.

4.2 Comparing Session Contents

The base statistics reflect the properties of the datasets of reconstructed sessions. However, they do not indicate to what extent the reconstructed sessions are the same as the real ones. The measures described in section 3 analyze session content in this way. Figure 1 shows the values of these measures for the two datasets. Results for M_{cre} are virtually identical to those for M_{crs}, and are therefore omitted.

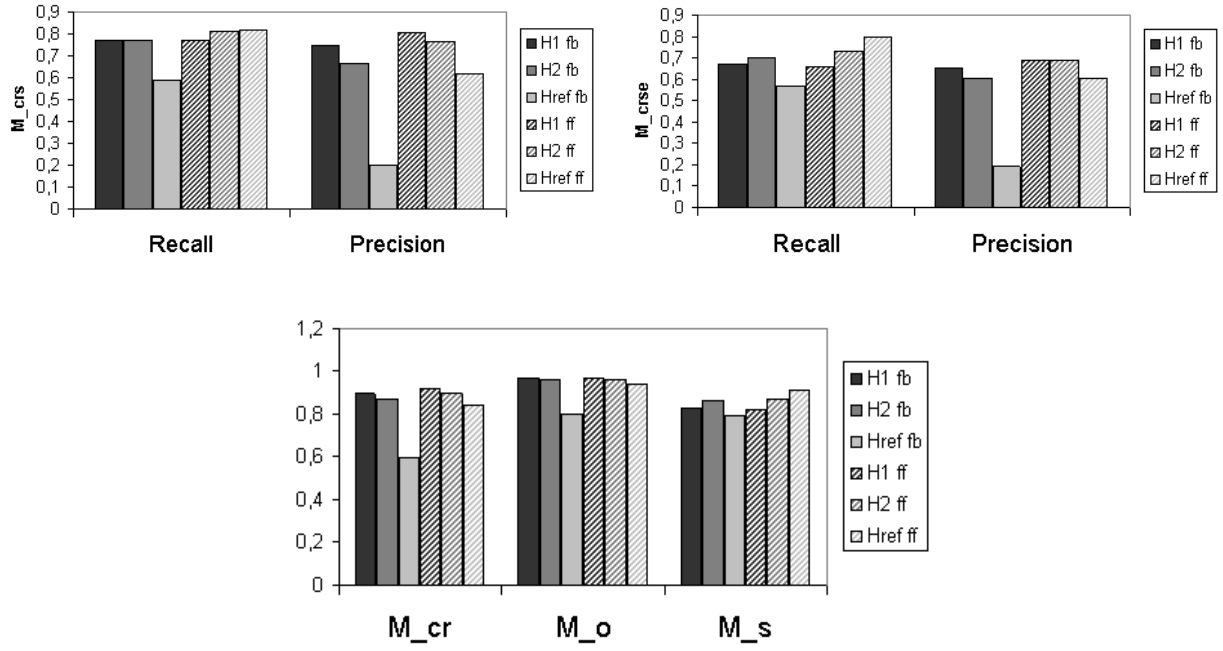

Figure 1: Recall and precision for the measures M_{crs} (top left) and M_{crse} (top right), and complete reconstruction and gradual measures (bottom), by heuristic and site structure: **h1** (black), **h2** (grey), and **href** (light grey); frame-based (solid), frame-free (hatched).

The quality of the reconstructed sessions with respect to our measures differs from heuristic to heuristic. Moreover, the presence of framesets does affect the results. In particular: (a) Heuristic **h1** has very similar scores for the frame-based and the frame-free version, and is thus "frame-insensitive". (b) Heuristic **h2** has higher scores for the frame-free version. (c) For the frame-based version, heuristic **h2** has best scores for the most restrictive measures (M_{crs}/recall, M_{crse}/recall, and M_s), while **h1** performs better for the less restrictive measures M_{cr} and M_o. However, its precision is very low. (d) Heuristic *hef* has much higher scores for the frame-free version than for the frame-based one. It outperforms the other two heuristics in the scores of the three categorical recall measures and on the gradual measure M_s. Its precision increases greatly relative to the frame-based version. Note that the similarity measure M_s shows the same behavior across site structures and heuristics as the recall measure M_{crse}. So in the frame-free site, **href** correctly identifies the highest proportion of real sessions in their entirety, and it retains the highest proportion of real session fragments.

In summary, heuristics **h1** and **h2** are most appropriate for sites providing both a frame-based and a frame-free version of their content. The good performance of **href** on the frame-free version suggests that it is appropriate if the sessions in the frame-free version are small.

4.3 The Impact of Session Length

Table 1 has shown that most of the real sessions are shorter than the average. Hence, it is of interest how effective each heuristic is in reconstructing the short sessions of the Web server log. Table 2 gives an overview of the distribution of session lengths.

	Proportion of sessions of this length in							
	$\mathcal{R}^{fb}$	$\mathcal{C}^{fb}_{\mathbf{h1}}$	$\mathcal{C}^{fb}_{\mathbf{h2}}$	$\mathcal{C}^{fb}_{\mathbf{href}}$	$\mathcal{R}^{ff}$	$\mathcal{C}^{ff}_{\mathbf{h1}}$	$\mathcal{C}^{ff}_{\mathbf{h2}}$	$\mathcal{C}^{ff}_{\mathbf{href}}$
1-page sessions	0.03	0.03	0.04	0.44	0.30	0.20	0.26	0.37
2-page sessions	0.02	0.03	0.04	0.09	0.15	0.17	0.16	0.15
3-page sessions	0.23	0.15	0.23	0.17	0.09	0.10	0.10	0.09
Larger sessions	0.72	0.79	0.69	0.30	0.46	0.53	0.48	0.38
Total no. of sessions	13829	14234	15971	41117	20950	20050	22149	27707

Table 2: Short sessions in the frame-based (*fb*) and in the frame-free (*ff*) site version

The measurements for the frame-based version show that the percentage of small sessions for the time-based heuristics is close to the percentage of real sessions of the same size. Only **href** produces a disproportionately large number of one-page sessions. In the frame-free version, **href** also produces a large number of one-page sessions, but the percentage of sessions produced for each length is similar to the percentage of sessions of this length in the log of real sessions.

	Proportion of real sessions with this length			
	in $\mathcal{R}^{ff}$	that were completely reconstructed in		
		$\mathcal{C}^{ff}_{\mathbf{h1}}$	$\mathcal{C}^{ff}_{\mathbf{h2}}$	$\mathcal{C}^{ff}_{\mathbf{href}}$
1-page sessions	0.30	0.53	0.71	0.93
2-page sessions	0.15	0.77	0.83	0.83
3-page sessions	0.09	0.77	0.80	0.82
4-page sessions	0.12	0.76	0.82	0.88
5-page sessions	0.08	0.75	0.78	0.84

Table 3: Completely reconstructed real sessions in the frame-free site version

Similarly to the base statistics, the statistics per session length do not guarantee that the real and the reconstructed sessions of length n are identical. For the frame-free version, we have computed the complete matches between real and constructed session, i.e. the percentage of real sessions that appear in the log of reconstructed sessions. The results for sessions with one to five pages are shown in Table 3. The first column repeats Table 2 for reference. The three proportions associated with the heuristics correspond to the value of the M_{crse} measure, as defined in section 3, but is taken over the sessions of a given length instead over the whole dataset of real sessions. Table 3 shows that the performance improvement of the referrer heuristic is due to the complete reconstruction of most small sessions.

These results indicate that the parameter setting for this heuristic favors small session lengths. To investigate this further, we varied **href**'s parameter Δ, i.e., the allowable time gap between consecutive log entries with undefined referrer within one session. Increasing Δ led to a decrease in the ratio of the number of constructed sessions to the number of real sessions. With Δ approaching 10 minutes, this ratio approached 2 for the frame-based site, and 1 for the frame-free site. Also, the impact of varying Δ in the range 0–30 seconds was much larger for the frame-based site. This is not surprising given that if a top frame page has an undefined referrer, then so do all the components.

With increasing Δ, the median duration of the constructed session converges towards the median of the

real sessions for large Δs; the median of the constructed session sizes remains constant while the average increases towards the average of real session sizes. This implies that larger Δs allow for the reconstruction of the few long sessions, too.

The top value of Δ that we investigated, 10 minutes, is equal to **h2**'s parameter δ, the maximum page stay time on *any* page. With Δ equal to δ, the two heuristics treat pages with undefined referrers alike, but page with defined referrers differently. **h2** splits at a page with a defined referrer if the elapsed time was too long. This bears the risk of mis-interpreting a page that really was inspected for a long time. **href** splits if the referrer was not in the current session. This bears the risk of mis-interpretation if, for example, a request was delivered from the cache. These two effects have to be investigated in their interaction in more detail, and a combined heuristic may be desirable.

5 Impact of user environment on session reconstruction quality

In this section, we focus on the influence of the user environment information concerning IP and agent. In an "ideal" setting, each user would have her own individual computer with a fixed IP address, and use the same browser for a longer time. This would make IP+agent equivalent to a cookie. However, in many real settings, the relation is not one-to-one.

To assess the impact of the user environment on sessionization heuristics, we considered two important sources of error in heuristic identification of unique users: (1) when a single user accesses the site with multiple IP+agent combinations, and (2) when one IP+agent combination is shared by multiple users. We compared (in the frame-free site) users identified through IP+agent to those identified using cookies.

(1) One user accesses the server with different IP+agent combinations. The most common reason for the use of different IP addresses is dial-up access to an ISP which assigns a dynamic IP address at each new connection. The use of different agents may occur because users upgrade their browser version, or when they use different browsers at different times. Therefore, IP+agent separation will generally construct more "pseudo-users" than there are "cookie-identified users".

Our log contained 5446 different cookies (and hence, by assumption, users). It contained 6849 unique IP addresses, and 8409 unique IP+agent combinations.

77.38% of all users never changed their IP address, and 96.49% never changed their agent. 75.98% never changed their IP+agent combination. Only 6.15% changed their IP+agent combination more than three times, which would represent the expected behavior of a frequent user of the site whose ISP assigns dynamic IP addresses. This may reflect the fact that 10-20% of users of the site are people working at their own desk on campus, or logging in from university computer rooms, i.e., from hard and non-shared IP addresses. However, the remainder access the site from the outside and can thus be considered representative of users of a broader range of site types.

More importantly, these errors do not propagate to the session level: Fewer than 5% of real sessions contained multiple IP+agent combinations. This suggests that IP+agent could be quite effective if the analysis is done at the session level (without regard to users). However, there could be a significant error if the analysis is to be done at the user level (e.g., for finding patterns among repeat users).

(2) One IP+agent combination is shared by several users. The most common reason for IP address sharing is the use of the same proxy server by different people. Also, the number of market-dominating browsers and versions is small, so that there are many people with identical user agents.

If the visits of different users from one IP+agent are not overlapping in time, and the temporal interval between their activities is long enough, all three heuristics will correctly introduce a session boundary. *Simultaneous* use of the same IP+agent combination by different users is the central problem introduced by the multiple use of IP+agent. Temporally interleaved sessions cannot be segmented correctly by a temporal heuristic. **href** can distinguish the trails of different, simultaneous users only if they access different objects from different referrers. This is not the most usual case: Rather, most users start at a major entry page of the site, which becomes the common referrer for the next access. Moreover, many object requests have an

undefined referrer. Some Web servers, including the IIS used in our test site, leave the referrer undefined under a large number of conditions.

In our log, 86.98% of IP addresses were used by only one, and only 2.8% by more than three users. These were mainly proxy servers, most of them AOL. The identification performance of IP+agent was better: 92.02% of IP+agent combinations were employed by only one user, and only 1.32% by more than three users. So 8% of IP+agent combinations involved accesses from different users. However, another test of our log showed that *simultaneous* accesses from different users with the same IP+agent combination accounted for only 1% of the log sessions. Thus, our log presents very good conditions for analysis: An IP+agent combination can be equated with one cookie / user in the large majority of cases.

While it would be interesting to split the log and thus experimentally determine the influence of the non-unique IP+agent–user relation on reconstruction quality, the scarcity of these cases, in particular of simultaneous sessions, means that this would probably not produce statistically meaningful results. We have therefore decided to make this the subject of further work with different data. The small percentage of simultaneous sessions from the same IP+agent may be peculiar to the university site we are studying. Commercial sites of e-shops, e-auctions or public administration are accessed by a much more diversified population of users, so that larger numbers of simultaneous sessions from the same IP+agent can be expected. The absence of a reliable assignment of activities to users is not easy to amend: As shown in [SMBN02], the three heuristics show a performance drop of circa 20% if the mapping of activities to users is solely based on the IP+agent combination.

6 Impact of session reconstruction on mining applications

In this section, we investigate the impact of session reconstruction on two applications: the analysis of frequent entry and exit pages, and the recommendation of pages based on previous visitors' usage. We relate the results to those on session reconstruction quality *per se*, and discuss further work.

6.1 Impact on entry/exit page analysis

This application is often the first step of Web site analysis, and it is very important for customer-oriented services. It gives insights concerning which pages are first seen; their quality determines whether the user will visit further pages. The exit page is one at which the user abandoned the site; if this leaving is not desired, then page redesign is necessary. Misclassifications of entry and exit pages lead to misinterpretations of user behavior and non-rewarding design efforts, and should therefore be avoided.

To evaluate the performance of the three reactive strategies, we again use the measures of *precision* and *recall*. In particular, let E be the set of entry pages in the real sessions, and let E_h be the set of pages characterized as entry pages by the heuristic h. Then, the *precision* of h is the ratio of pages correctly classified as entry pages to all pages characterized by the heuristic as such, while *recall* is the ratio of correctly classified entry pages to all real entry pages: $precision(h) = (|E \cap E_h|)/(|E_h|)$, and $recall(h) = (|E \cap E_h|)/(|E|)$. For exit pages, precision and recall are defined in the same way. The results for entry pages are shown in Fig.2. The results for exit pages are virtually identical.

The different performance of the heuristics can best be explained by relating these values to the number of sessions constructed by them. In particular, **href** constructed very large numbers of sessions, including large numbers of one-page sessions, which turned almost all pages in the site into entry pages. This is likely to find all real entry (exit) pages, creating high recall values. But since it erroneously considers so many pages to be entry (exit) pages, precision is low. The reverse holds for the temporal heuristics, which tend to reconstruct sessions more faithfully.

The breaking up of sessions and the consequent overestimation of the number of sessions was less pronounced in the frame-free site. This was particularly so for **href**. Therefore, fewer incorrect guesses of entries/exits are made, and precision increases. Note that recall performance did not suffer.

In summary, **h1** and **h2** proved to be robust heuristics, both with respect to site structure and in terms of the comparison between their (high) recall and precision values, with a tradeoff between a somewhat higher

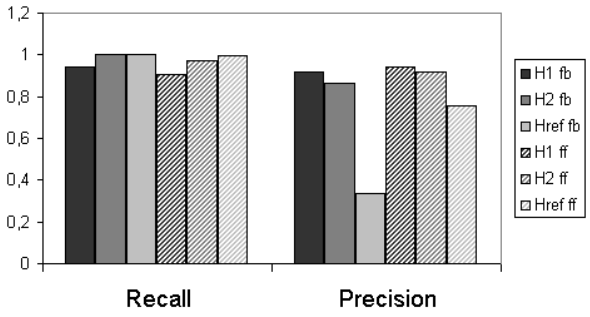

Figure 2: Recall and precision for entry page analysis by heuristic and site structure: **h1** (black), **h2** (grey), and **href** (light grey); frame-based (solid), frame-free (hatched).

precision (**h1**), or a somewhat higher recall (**h2**). The precision of **href** is generally lower, and it is strongly affected by the presence of frames. However, it has a very high, and robust, recall performance.

These findings closely parallel the results reported in section 4.3; witness in particular the structure of results for the frame-free site and for precision values between Figs. 1 and 2. The main difference between these figures is the relative performance of **href** on recall: It reconstructs fewer real sessions correctly than **h1** and **h2** do. This is a consequence of the segmenting behavior of **href**, which will lead to a high probability that all real entry pages are identified, but will then erroneously cut many of them too soon to capture the remaining real session. This indicates that results on the impact of factors like site structure on the quality of session reconstruction by different heuristics also give information on the impact of these factors on the quality of subsequent mining results.

6.2 Impact on page prediction/recommendation

Recommender systems are another relevant mining application that depends heavily on the correct reconstruction of sessions. Based on co-occurrences of pages found in previous sessions, a new visitor in an ongoing session can be given recommendations for pages that are likely to be interesting to her given the pages visited so far in her ongoing session. One method of determining co-occurrences is based on the clustering of sessions, e.g., [MCS00].

Mobasher, Dai, Luo, and Nakagawa [MDLN02] have proposed a method of evaluating the quality of different clustering methods according to their profiles' predictive power and recommendation quality. In particular, they tested PACT (Profile Aggregations based on Clustering Transactions) against two other methods for clustering and profile creation. PACT was also used here: The transactions are sessions consisting of visits to pageviews p from a set of pageviews P, expressed as vectors of $\langle p, weight \rangle$ pairs. In the current experiments, the weight of a pageview specifies whether it was visited in this session (1) or not (0). The transactions were clustered using k-means, and for each transaction cluster c, the centroid (mean vector) was computed. The weight of a pageview in the mean vector is the proportion of sessions belonging to this cluster in which this pageview was visited. A threshold of 0.7 was then applied,[1] and the result constituted the *profile* pr_c of that cluster: $pr_c = \{\langle p, weight(p, pr_c)\rangle \mid p \in P, weight(p, pr_c) \geq 0.7\}$. This can be represented as a vector in the original space, by using weights of zero for the other pageviews.

Predictive power was measured by the "weighted average visit percentage", WAVP. This allows us to evaluate each profile individually according to the likelihood that a user who visits any page in the profile will visit the rest of the pages in that profile during the same session. It is computed by calculating the average similarity of a profile to each session (their scalar product), averaging over sessions, and normalizing by the sum of weights in the profile [MDLN02].

In [MDLN02], WAVP was used to compare different sets of profiles obtained using different methods of

[1] The results were similar for other values ≥ 0.5.

clustering, in order to evaluate the quality of these methods. Here, we used the same method of clustering throughout (PACT), and compare different sets of profiles obtained from different sets of (real or constructed) sessions.

We used data from the frame-free setting, with the standard parameter values (see section 2.2).

In our experiment, we concentrated on the error introduced by session construction. We therefore used a common baseline for testing prediction quality: the profiles obtained by clustering the set of real sessions. The WAVP values of applying these profiles to these sessions constitute the best possible baseline. Then, the constructed sessions produced by the different heuristics were clustered to obtain heuristic profiles, and WAVP values of applying these profiles to the original set of real sessions were computed.

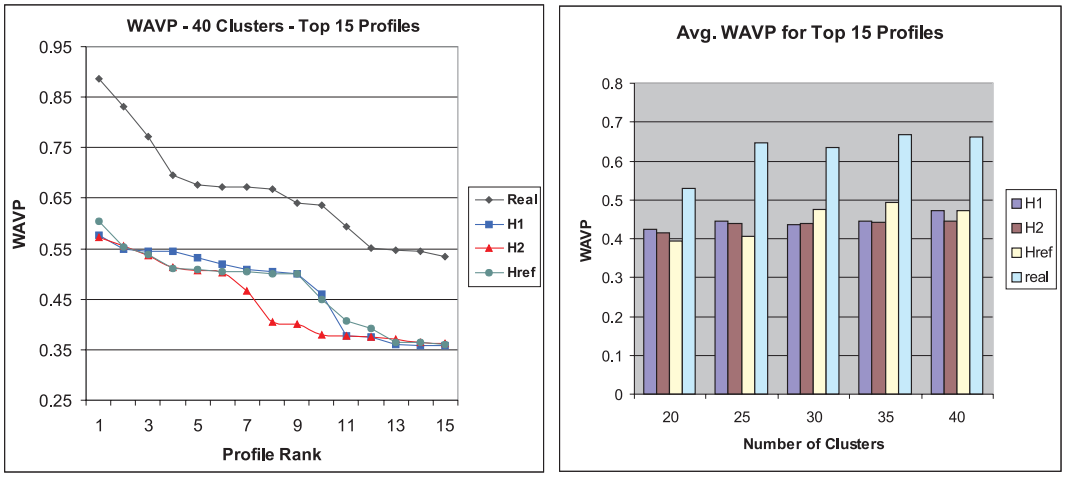

Figure 3: WAVP of the top 15 profiles: (a) WAVP values at $k = 40$: baseline (real) and profiles built from constructed sessions (**h1, h2, href**). (b) Average WAVP values for different k.

This procedure was repeated for different values of k. The best values for all ways of session (re)construction were obtained for the number of clusters k between 35 and 40. Figure 3 (a) shows the quality of prediction for the top 15 profiles at $k = 40$, ordered by their WAVP value. The differences for lower-ranking profiles are negligible. It shows that the impact of heuristic session reconstruction (the difference between the "real" and "heuristic" WAVPs) is approximately constant for all profile ranks. It also shows that while the difference in performance between heuristics is not large, **href** and **h1** have the highest values and are (near-)identical for most ranks. Figure 3 (b) shows the average WAVP, depending on the number of clusters k. This shows a slight superiority of **href** for $k \geq 30$.

A possible explanation for this result is again the tendency of **href** to construct short sessions, and thus short profiles. These short sessions capture a large percentage of real session entries (see Fig. 2) and probably also the next few pages. In the frame-free site, they also capture the largest percentage of real session content (see Fig. 1). In addition, profile construction selects only the most frequent parts of these short constructed sessions. So many real sessions will contain the requests for the pages in the profiles. Thus, the average similarity and WAVP will be high, in particular for the top ranking profiles that capture the most typical behavior. **href** profiles will tend to generate fewer, but safer, recommendations. In contrast, the prediction quality of **h1** profits from its rather faithful reconstruction of sessions. (The conclusion regarding **href**, however, will likely not hold up if the site is highly frame-based.)

Taken together, these results indicate that for prediction, **href** and **h1** perform rather well. Future work will include further analysis of recommendations based on the different heuristics for their "usefulness", or the extent to which they include "interesting" items, see [MDLN02] for a methodology.

7 Conclusions

In this study, we have compared the performance of session reconstruction heuristics. We expect that performance is related to the predictive power of the sessions built by the heuristics. Hence, we have compared the quality of predictions for entry/exit pages and of recommendations based on usage profiles. Since session reconstruction heuristics do not necessarily produce the dataset of *real* sessions, we have established and conducted a suite of experiments that compare the datasets of real and of constructed sessions in terms of base statistics, contents and distribution of session lengths.

An essential aspect of our investigation has been the comparative study of the impact of framesets. To this purpose, we have compared our findings on server logs from the frame-based and the frame-free version of a site. Although one cannot claim that any site is representative of all sites in the Web, our experiments indicate that the presence of framesets does not affect all heuristics uniformly. In particular, time-based heuristics are less affected by the presence of framesets, while the referrer heuristic exhibits very poor performance. On the other hand, this same heuristic improves dramatically when reconstructing small sessions in the frame-free site.

An essential application of reactive heuristics is the reconstruction of sessions comprised of page views in multiple servers. Even in the presence of cookies, the synchronization of the cookies or of generated session identifiers is not always possible. Our results indicate that the referrer heuristic, which is per se designed for this kind of application, may perform satisfactorily in frame-free sites with small sessions only. Since the other heuristics exhibit better performance, we intend to investigate how combinations of reactive heuristics perform on the union of logs of multiple servers.

We have studied the predictive power of the constructed sessions for the prediction of entry/exit pages and for recommendations based on usage profiles. Much work in web usage analysis is focusing on the establishment of such profiles, i.e. on descriptive patterns. We intend to elaborate on the impact of session reconstruction on the quality of clusters by establishing an appropriate comparison framework.

References

[BMSW01] Berendt, B., B. Mobasher, M. Spiliopoulou, J. Wiltshire. 2001. Measuring the accuracy of sessionizers for web usage analysis. *Proceedings of the Workshop on Web Mining, First SIAM International Conference on Data Mining, Chicago, IL.* 7–14.

[BS00] Berendt, B., M. Spiliopoulou. 2000. Analysis of navigation behaviour in web sites integrating multiple information systems. *The VLDB Journal* **9** 56–75.

[CP95] Catledge, L., J. Pitkow. 1995. Characterizing browsing behaviors on the world wide web. *Computer Networks and ISDN Systems* **26** 1065–1073.

[CMS99] Cooley, R., B. Mobasher, J. Srivastava. 1999. Data preparation for mining world wide web browsing patterns. *Journal of Knowledge and Information Systems* **1** 5-32.

[MCS00] Mobasher, B., R. Cooley, and J. Srivastava 2000. Automatic personalization based on web usage mining. *Communications of the ACM* **43(8)** 142–151.

[MDLN02] Mobasher, B., H. Dai, T. Luo, and M. Nakagawa 2002. Discovery and evaluation of aggregate usage profiles for Web personalization. *Data Mining and Knowledge Discovery* **6** 61–82.

[PZK01] B. Padmanabhan, Z. Zheng, and S. O. Kimbrough. Personalization from incomplete data: What you don't know can hurt. In *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2001)*, pages 154–163, San Francisco, CA, August 2001.

[SF99] Spiliopoulou, M., L.C. Faulstich. 1999. WUM: a tool for Web utilization analysis. *Proceedings EDBT Workshop WebDB'98*, LNCS 1590, Springer, Berlin, Germany. 184–203.

[SMBN02] Spiliopoulou, M., Mobasher, B., Berendt, B., & Nakagawa, M. A framework for the evaluation of session reconstruction heuristics in Web usage analysis. To appear in *INFORMS Journal on Computing*.

[W3C99] World Wide Web Committee Web Usage Characterization Activity. 1999. *W3C Working Draft: Web Characterization Terminology & Definitions Sheet.* www.w3.org/1999/05/WCA-terms/

[ZPK02] Zheng, Z., B. Padmanabhan, and S. Kimbrough. On the existence and significance of data preprocessing biases in Web usage mining. To appear in *INFORMS Journal on Computing*.