LumberJack: Intelligent Discovery and Analysis of Web User Traffic Composition

Ed H. Chi, Adam Rosien, Jeffrey Heer

PARC (Palo Alto Research Center)
3333 Coyote Hill Road
Palo Alto, CA 94304
{echi, arosien, jheer}@parc.com

Abstract. Web Usage Mining enables new understanding of user goals on the Web. This understanding has broad applications, and traditional mining techniques such as association rules have been used in business applications. We have developed an automated method to directly infer the major groupings of user traffic on a Web site [Heer01]. We do this by utilizing multiple data features in a clustering analysis. We have performed an extensive, systematic evaluation of the proposed approach, and have discovered that certain clustering schemes can achieve categorization accuracies as high as 99% [Heer02b]. In this paper, we describe the further development of this work into a prototype service called LumberJack, a push-button analysis system that is both more automated and accurate than past systems. **Keywords**: Clustering, Log Analysis, Web Mining, User Profile, User Sessions, World Wide Web

1 Introduction

The Web has become part of the fabric of our society, and accordingly we have an increasing need to understand the activities and goals of web users. We can improve nearly every aspect of the user experience on a Web site by understanding the users' goal and traffic composition. Webmasters and content producers would like to gain an understanding of the people that are visiting their Web sites in order to better tailor sites to user needs [Yan96]. Marketers would like to know the users' interests in order to have better sale promotions and advertisement placements [Barrett97]. News sites would like to produce and present materials that are highly relevant to their visitors.

Traditional user activity analysis methods such as user surveys are labor-intensive and intrusive when employed daily, slow to apply to on-the-fly Web personalization, and inaccurate due to surveying response inconsistency. Instead, what is needed is an automated means of directly mining the Web server logs for groupings of significant user activities. Web Usage Mining, and more specifically, user session clustering, is a relatively new research area [Cooley97, WEBKDD01, SIAM01], in which user profiles are extracted from the server logs and then grouped into common activi-

ties such as “product catalog browsing”, “job seeking”, and “financial information gathering”.

Web Mining techniques build user profiles by combining users’ navigation paths with other data features, such as page viewing time, hyperlink structure, and page content [Heer01, Srivastava00]. While the specific techniques vary [Shahabi97, Fu99, Banerjee01, Heer01], the end goal is the same: to create groupings of user sessions that accurately categorize the sessions according to the users’ information needs.

There are two major issues with existing approaches. First, most approaches examine one or two combinations of data features for clustering the user sessions. What is needed is an approach that allows for any of the data features to be used, as the situation dictates. For example, sometimes page viewing time might not be available, or page content may be too expensive to gather and analyze, so the user session clustering techniques have to be adaptable to these situations.

Second, in the literature, each technique’s validation is conducted on a different Web site, making it extremely difficult to compare the results. We have no basis from which to choose one data feature over another. What’s worse is that, since only user traces are used, there is no way of knowing a priori what the true user information need is for each user session. So we had no way of knowing whether the algorithms performed correctly and clustered the sessions into appropriate groupings.

In this paper, we describe our prototype production system code-named LumberJack that integrates all of this information into a single report that analysts can use in the field to accurately gain an understanding of the overall traffic patterns at a site. We first summarize our previous work on solving both of these issues, then we describe case studies of LumberJack in action in the real world. We also describe the challenges and problems that we have encountered in practice.

First, we describe our development of a technique that combines any of the available data features to cluster user sessions [Heer01]. The idea of combining multiple features in the context of k-Means clustering is not new [Schuetze99b, Heer01, Modha02]. By integrating these data features into a single system, we can more intelligently pick and choose any of the data features to utilize.

Second, to analyze the effectiveness of the proposed clustering data features, we gathered user sessions for which we know *a priori* the associated information goals, thus enabling us to evaluate whether the clustering algorithms correctly categorized the user sessions [Heer02b]. We present a user study and a systematic evaluation of clustering techniques using these different data features and associated weighting schemes. We first asked users to surf a given site with specific tasks. We then use this a priori knowledge to evaluate the different clustering schemes and extract useful guidelines for Web usage analysis.

In the following two sections, we first describe our clustering approach, and then we describe the experiment that evaluated the precision of 320 different clustering schemes. Next, we discuss LumberJack, an automated web analysis service that combines both user session clustering and traditional statistical traffic analysis techniques. Two case studies illustrating the system’s efficacy are presented. Finally, we discuss some of the practical problems encountered and insights gained.

2 Method

The uniqueness of our approach is two-fold. First, our approach to user modeling is motivated by Information Foraging theory [Pirolli99b], and in particular the theoretical notion of Information Scent [Chi00, Chi01]. Information Scent is the user's perception of the value and cost of accessing a piece of information. Applying the notion in this context, we assume implicitly that what a user sees is a part of that user's information interest. Accordingly, we believe that combining the content viewed by the user with their actual navigational path is key to creating models that can accurately infer user's information needs, essentially using information cues to infer user goals [Chi01]. Second, we employ a modular system that combines multiple data features of each web page, in an approach called Multi-Modal Clustering (MMC), to construct user profiles. Some data feature combinations are novel; for example, past approaches do not use linkage structure in the clustering.

To create the analysis of user traffic, we first model each page of a website in a multi-feature vector space, utilizing page features such as the words, URL, inlinks, and outlinks. This creates a model of all of the Web pages that users accessed. We then construct a vector space model of user sessions as weighted combinations of the page vectors, using attributes of the users' sessions to determine the weights. We then define a similarity metric for comparing these user sessions and use it to generate the resulting clusters. Figure 1 below illustrates the steps of this process. We now describe each of these steps in greater detail.

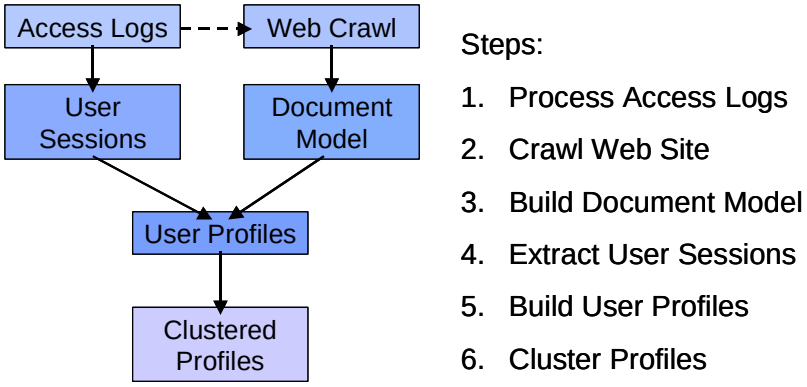

Figure 1 Method Data Flow – Analysis can be characterized by six stages.

The system begins in **Step 1** by processing web usage logs, typically in the form of standard web server logs. The logs are initially filtered to extract the relevant URLs. Requests for images and non-existent documents as well as all requests generated by web crawlers are removed. In **Step 2**, the content of the web site is retrieved using a web crawler. Any relevant pages listed within the access logs that were not captured by the basic web crawl are retrieved, allowing the system to capture dynamically generated pages such as search results.

In **Step 3**, the retrieved content is used to build a vector space model of the pages on the site. The system currently models four feature spaces in its representation: Content, URL, Outlinks, and Inlinks. Each feature space is called a *modality*. While these modalities are not exhaustive, they do provide good coverage of the textual and navigational content of the site. For example, one can imagine other interesting possibilities like visual page features such as color histogram data, banner ad placements, and navigational structures. We now describe each modality in more detail:

Content: The **Content** subvector of a page is a weighted keyword vector modeling the words that the user sees on that page. The words visible to the user are extracted, stop words are removed, and the remaining words are run through Porter's stemming algorithm [Porter80].

URL: The **URL** subvector of a page is a URL token keyword vector modeling the URL of the page. Each URL is tokenized using '/', '&', '?' and other appropriate delimiters.

Inlink/Outlink: The **Outlink** subvector of a page describes which pages are reachable from this page, while the **Inlink** subvector describes which pages link to this page.

These individual subvectors are then concatenated to form a single multi-modal vector $\boldsymbol{P}_d$ = ($\boldsymbol{Content}_d$, $\boldsymbol{URL}_d$, $\boldsymbol{Inlink}_d$, $\boldsymbol{Outlink}_d$) for each document *d*. The entire vector is represented in sparse vector format. Of course, in practice we can always pick and choose which modalities we want to include in the model.

We also selectively apply the Term Frequency by Inverse Document Frequency (TF.IDF) weighting scheme on the modalities. A common technique in the information retrieval field, TF.IDF provides a numerical value for each item in a document, indicating the relative importance of that item in the document. This weighting is roughly equal to an item's frequency in a given document divided by the frequency of the item occurring in all documents. Formally, TF.IDF can be expressed as $TF.IDF_{t,d} = \log(1 + tf_{t,d}) * \log(N / df_t)$, where $tf_{t,d}$ indicates the number of times an item *t* occurs in a given document *d*, and df_t indicates the number of documents in which the item *t* appears, and *N* indicates the total number of documents [Scheutze99a, p. 542]. In practice, we have found that TF.IDF works nicely in the content and URL modalities, and poorly in the linkage modalities.

In **Step 4**, we construct the user session model by representing the individual user sessions as vectors. We first sessionize the usage logs using standard techniques [Pirolli99a] and then represent each session as a vector $\boldsymbol{s}$ that describes the sessions' page views. We have explored a number of possibilities for weighting the pages in a session $\boldsymbol{s}$. These **path weightings** possibilities are:

(a) *Frequency*: Each page receives weighting equal to the number of times it is accessed, e.g. for a user $\boldsymbol{i}$, A→B→D→B, $\boldsymbol{s}_i$ = (1,2,0,1,0).

(b) *TF.IDF*: Treating each session as a document and the accessed pages as the document terms, each page receives a TF.IDF weighting.

(c) *Linear Order (or Position):* The order of page accesses in the session is used to weight the pages, e.g. A→B→D , $\boldsymbol{s}_i$ = (1,2,0,3,0).

(d) *View Time*: Each page in the session is weighted by the amount of viewing time spent on that page during the session, e.g. A(10s) → B(20s) → D(15s), $\boldsymbol{s}_i$ = (10,20,0,15,0).

(e) *Various Combined Weighting*: Each page in the session is weighted with various combinations of the TF.IDF, Linear Order, and/or View Time path weighting. For example, using both Linear Order+View Time: A(10s) → B(20s) → D(15s), $\boldsymbol{s}_i$ = (10,40,0,45,0).

In **Step 5**, we combine the document and session models to construct a set of user profiles. We assume implicitly that each page a user sees is a part of that user's information interest. Each user profile $\boldsymbol{UP}_i$ is constructed as the linear combination of the page vectors $\boldsymbol{P}_d$ scaled by the weights in $\boldsymbol{s}_i$. That is,

$$UP_i = \sum_{d=1}^{N} s_{id} P_d$$

In order to do comparison between user profiles, we need to define a similarity metric ***d()*** over the profiles. Our system uses the standard cosine measure, which measures the cosine of the angle between two vectors, applied to each modality individually. The values of these comparisons are then linearly combined to obtain the resulting similarity value. Because we use the cosine measure, each user profile must first undergo normalization, where each modality subvector is normalized to unit length. We can formally represent the similarity measure as:

$$d(UP_i, UP_j) = \sum_{m \in Modalites} w_m \cos(UP_i^m, UP_j^m) \qquad \sum_m w_m = 1$$

By adjusting the **modality weights** w_m, we can specify the relative contribution of each modality in the similarity function, for example by weighting page content higher than the other modalities.

Finally, in **Step 6**, using this similarity function we can then proceed with clustering. We use a recursive bisection approach as described in [Zhao01], which starts by placing all user profiles in one initial cluster and then repeatedly bisects clusters using the traditional K-Means algorithm [MacQueen67] until a specified number of clusters *k* is achieved. The corresponding criterion function maximizes the within cluster similarity between members of a cluster *S* and the cluster's centroid *C*:

$$\max \sum_{r=1}^{k} \sum_{UP_i \in S_r} d(UP_i, C_r)$$

This criterion has proven very successful within the context of document clustering [Zhao01], and so we apply it here with high expectations, as our user profiles are in fact document aggregates. Though not reported here, we have indeed found this

criterion to work best among a number of clustering techniques. Details of this specific approach can be found in [Zhao01].

In previous work [Heer02a], we have explored using a stability-based method [BenHur02] for finding the optimal number of clusters. Though the method displayed some promise, overall our results proved inconclusive. As we collect more usage data sets in the future, we hope to settle on an acceptable scheme.

In summary, we model both the web site and the collected usage data, and then combine these models to generate a set of user profiles represented as multi-modal vectors. We then cluster these profiles to obtain categorizations of the user sessions.

3 Evaluation

We recently performed a systematic evaluation of different clustering schemes using the above approach by conducting a user study where we asked users to surf a large corporate site with *a priori* specified tasks [Heer02b]. That is, we specify to the users what their goals should be, and ask them to surf according to that user intent. By knowing what the tasks were and how they should be grouped in advance, we were able to do post-hoc analysis of the effectiveness of different clustering schemes. Here we describe this experiment briefly.

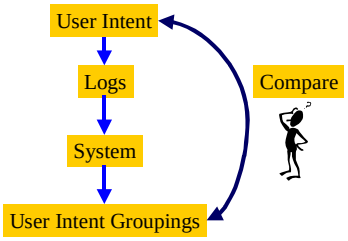

Figure 1: By conducting a user study where we give the users specific tasks, we know *a priori* how the tasks should have been grouped. In the experiment, we run the user traces through the system, and compare the results of the system analysis with the *a priori* groupings.

We asked 21 participants to surf Xerox.com and captured their clickstreams using the WebQuilt proxy [Hong01]. There were a total of 15 tasks, organized into five information need groupings consisting of three tasks each. Individual tasks were chosen using email feedback from the site. The five task groups were “product info”, “support”, “supplies”, “company info”, and “jobs”. We assigned the tasks randomly, but with each task assigned roughly the same number of times. Subjects were asked to perform the tasks in their natural browsing environment and allowed to start and stop a task anytime they wanted. We collected a total of 104 user sessions.

We studied a total of 320 different algorithm schemes using various modality weighting and path weighting methods in combination. As depicted in Figure 1, we then measured accuracy by counting the number of correct classifications (comparing

against our *a priori* task categories) and then dividing by the total number of sessions.

We discovered that, by counting the number of correct categorizations, certain combinations of data features enabled us to obtain accuracies of up to 99%. The traditional scheme of clustering just the user session vectors gives an accuracy of only 74%, while certain combinations give accuracies below 60%.

Two trends were significant. First, Content performed admirably, with poor performance only when used with the TF.IDF path weighting alone. Figure 2a shows how content outperformed other uni-modal schemes. A Linear Contrast showed that content performed significantly better than the other four uni-modal schemes ($F(1,35)=33.36$, $MSE=.007332$, $p<0.0001$). Expanding this to all multi-modal schemes, we compared all of the content-based schemes vs. non-content-based schemes. This contrast was also significant ($F(1,105)=32.51$, $MSE=.005361$, $p<0.0001$). Thus, crawling the site and using the page content to help cluster user sessions greatly increases algorithm accuracy. This is far from surprising; intuitively, the words that the user sees during each session are good indicators of their information need.

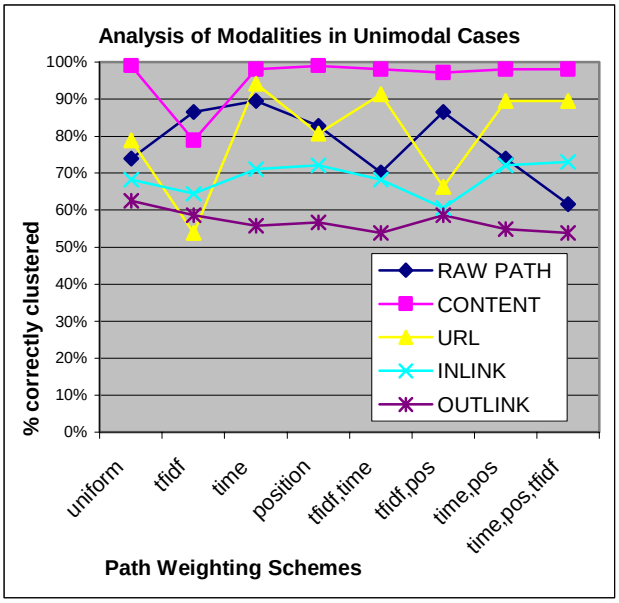

Figure 2: (a) Plot of each different modality's accuracies across different path weighting schemes.

Second, we found that the viewing time spent at each page is a good data feature for clustering the user sessions. Figure 2b shows the analysis of path weighting. The left portion of the chart shows the uni-modal cases, while the right side shows the multi-modal cases. What's most interesting from this chart is that the View Time path weighting performed well, staying at the top of the curve across the chart. This is true regardless whether we're looking at the uni-modal or the multi-modal

schemes. A paired t-Test found a significant difference between View-Time-based schemes vs. non-View-Time-based schemes (n=60, *V.T.mean*=89.5%, *s.d.*=12.7%, *non-V.T.mean*=83.2% *s.d.*=12.0%, *t(59)*=4.85, *p*=4.68e-6).

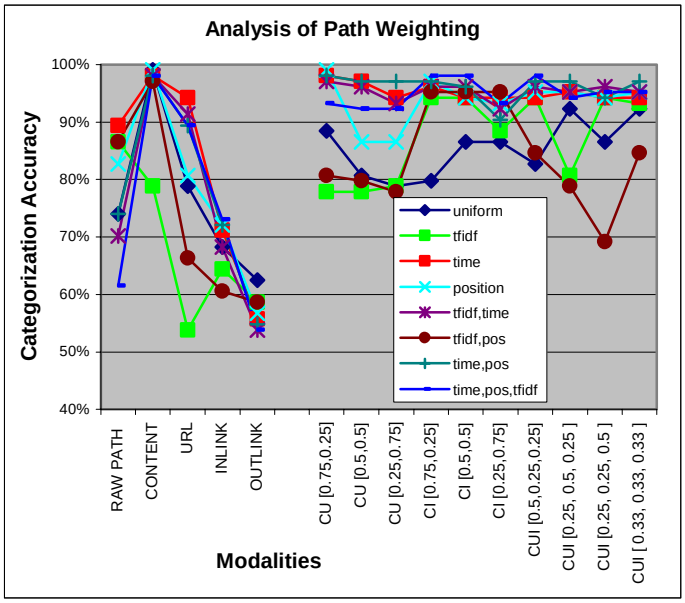

Figure 2: (b) Plot of each path weighting schemes against modality. (Modality shorthands are C=Content, U=URL, I=Inlink, O=Outlink)

We also discovered that multi-modal schemes are more robust in real life applications. There are many Web sites that have pages containing only images, sounds, videos, or other media formats. These pages cannot be parsed for content, making the usage of other modalities much more important. We could tailor the clustering technique to the unique features of the site being analyzed. For example, if the site has many pages without word content, then Inlink and URL Token modalities could be used in combination with View Time.

4 LumberJack Log Analyzer

Encouraged by the successful evaluation, we have applied our approach to a new service, code-named LumberJack, that is designed to remove the responsibility of the analyst to "chunk" behaviors, creating a robust starting point for the analyst to begin understanding the behavior of each type of user. Using the techniques described above, LumberJack processes Web server logs and automatically crawls the content and hyperlinks of the site to construct a model of user activity. It performs the clustering analysis described previously (in part using George Karypis' CLUTO toolkit

[CLUTO02]), and then computes a number of statistical analyses for *each* discovered user group, in contrast to statistics about the entire population shown in typical user activity reports. The LumberJack system is designed to be a push-button operation, in that only the server logs are needed, and the rest of the system runs completely automatically.

One motivation for the LumberJack system has been a lack of robust identification of user needs in traditional traffic analyses. For example, it may be known that users travel from a product list to a specific product 12% of the time. What isn't known is under what specific circumstances this aggregate behavior occurs; perhaps there are two classes of users, one that knows exactly what type of product they want, the other who are only interested in browsing many products to understand the available options. An analyst is required to infer these circumstances herself based upon little or no information. They may sort through data manually or make guesses, but user session clustering can provide a solid foundation an analyst may rely upon. Conversely, solely discovering user groups isn't necessarily actionable for an analyst or site designer either, as the specific statistics about a group are missing. Analyses of user needs and traffic are most useful when paired together.

The output of the LumberJack system is a web-based report, describing the different user groups and their properties. Table 1 summarizes the various cluster statistics available in a given report.

Report detail	Statistics	Benefit
User group summary	For each group: Number of sessions, percentage of population, Keywords that best describe the user group, Cluster quality (internal/external similarity)	Segments users by need/activity
Most frequent documents	Top ten most frequently occurring documents: Absolute and relative Time spent (mode	Where are users going? Are users focused on key pages, or do they have broad interests?
Representative paths	Sequences of pages that best describe typical activity (user sessions nearest cluster centroid)	Reduces session population to show best examples
Session path length trends	Histogram of user session path lengths	Characterizes "Law of Surfing" behavior
Session time trends	Histogram of user session times	Characterizes "Law of Surfing" behavior
Session document co-occurrence	How often the most frequent documents appear in the same session for all the sessions in the group	Shows task success rates
Session document transitions	How often the most frequent documents are followed by another in a session, how often each document is the first or last document in a session	Characterizes attrition rates between pages Detect "pogosticking", i.e., repeated sequences of forward and backward transitions like product lists to product detail pages

Table 1: Summary of LumberJack Report Details

Figure 3 illustrates a sample LumberJack report for a consumer information web site about diamonds. The current system generates static reports for a pre-specified number of clusters. In the future we plan to extend the reports to support interactive exploration of the generated cluster hierarchy, allowing analysts to refine their focus to particular clusters of interest, viewing user goals at multiple levels of granularity.

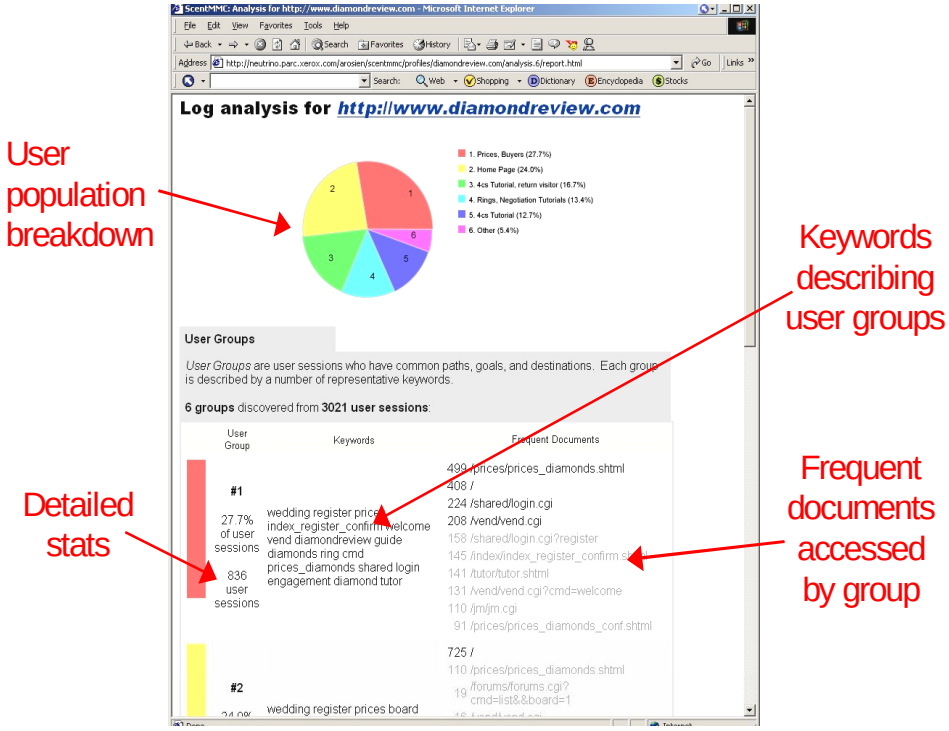

Figure 3: Sample LumberJack report for DiamondReview.com, Jan. 4-8, 2002

When used to analyze the site DiamondReview.com (Figure 3), LumberJack discovered six user groups in 3021 user sessions for 4 days starting from Jan. 4, 2002. 27.7% of the users were buyers interested in prices, while 24.0% spent most of their time on the home page. 16.7% were return visitors going through a diamond tutorial, and we know this because they had obviously started the tutorial from a half-way point that they have bookmarked. 13.4% were ring buyers, and 12.7% were first time tutorial readers. The remaining 5.4% were users that looked at many different pieces of information. Additionally, of the return tutorial users, the median session length was 11 clicks, mid-way into the tutorial, and at each step of the tutorial, the attrition rate was only 10%. It was around 35% for first time tutorial viewers; we hypothesize that those users who returned would have a greater impetus to continue the tutorial and finally making a purchasing decision.

Our report directly impacted the company in one significant way. We discovered that most users that starts the diamond tutorial finish reading the tutorial, even though it takes 30 minutes to finish going through the entire tutorial eventually. The designers of the site had anticipated that most users would not choose to finish the tutorial, so a link to diamond prices was placed at every step of the tutorial as an exit point, except on the last page. Since most users finish the tutorial, we recommended to Diamondreview to include purchasing information at the end of the tutorial.

We also studied using our tool on very large corporate sites. Figure 4 shows the result of running this analysis tool on the access logs from Xerox.com on July 24, 2001. The most striking result is that a large segment of user sessions (41.4%) center around the splash page. Viewing the actual clustered sessions revealed that these sessions consisted primarily of the site's splash page, with many paths jumping between the splash page and other linked pages. This could indicate that a large segment of users may come to the site without well-defined information needs and/or that the site may suffer possible usability problems that prevent users from successfully moving deeper into the site.

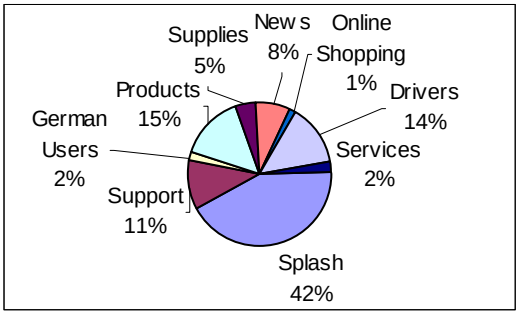

Figure 4: User Groups discovered for Xerox.com, July 24, 2001

Other substantial groupings include Xerox Product Catalog browsing (15%), Driver downloads (13.9%), Technical Support (11.5%), and Company News and Information (8.2%). One unexpected result was that there was a strong, concentrated group of German users that necessitated a unique cluster (1.7%). Xerox Sales and Marketing might also be interested to know the number of Online Shopping / Purchase related sessions (1.3%) in comparison to the number of product catalog viewers. As discussed in [Heer01], more detailed information about these groupings can be obtained by reclustering a given cluster. For example, we learned that within the Products group, 43% of the sessions centered around the Phaser line of printers. This information could be extremely useful to the marketing department, who can more carefully place advertisements of the Phaser printers at strategic routes common to the Products group.

In summary, LumberJack automatically analyzes the user traces obtained from a Web server log and clusters user sessions into significant groupings that describe the traffic composition at a Web site. It then generates reports describing the properties of these groupings, helping analysts understand the various activities of users on their

sites. These case studies illustrate that these user session clustering methods are indeed applicable to real world situations, and are scalable to large, heavily used Websites.

5 Difficulties and Lessons Learned

While developing our analysis techniques and building the LumberJack system, we have encountered a number of practical issues. Crawling and parsing a wide variety of web sites currently presents the greatest challenges. For example, sites may dynamically create pages with unrepeatable URLs that cannot be retrieved later, or pages containing dynamic links generated by JavaScript. Sometimes breadth-first traversals of a site (typical of most crawlers) may not be possible because of business logic requiring sequences of pages to be accessed in a particular order. Indeed, we have encountered many other technical issues.

However, the multi-modal nature of LumberJack's user profiles provides a method to minimize the effects of one or more deficient modalities. For example, if links cannot be resolved to valid URLs of a crawl, one may trade the use of the inlinks modality for another. The necessity of trade-offs will hopefully be reduced in the future by increased technical sophistication of crawlers and parsers, greater standardization of web site designs and logic, or more clever solutions such as the instrumentation of proxy servers. Moreover, the choices of which modalities to use may be automated by approaches similar to Caruana and Freitag's work on using hillclimbing to do feature selection [Caruana94]. They were able to obtain good results in Mitchell's Calendar APprentice Project.

In practice, the high dimensionality of the vectors must be represented very efficiency using sparse vector format. We use the existing compaction techniques for this purpose. Also, while we were able to obtain accurate results using k-Means in our laboratory studies and in real-world situations, we are not entirely certain that this method can tackle all of the complex sites out there with varying user tasks. Further studies are needed to understand the applicability in a wide variety of situations. Existing literature suggest that k-Means is often unable to tackle complex high dimensional spaces with low cluster separations. However, since our use of k-Means is only limited to doing bisections, this problem might be minimized. We suspect that with complex sites a graph partitioning method might be more appropriate and more robust.

Finally, our unique approach to the problem is deeply rooted in our user-centered point of view. We have designed the mining method from the ground-up to use the information cues that a user encounters as she surfs from one page to another. We use these information cues to try and understand how this conveys her information goals. Our experimental study is also rooted in this unique approach by evaluating it using user data that we know is flawless. We have written extensively on this Information Scent approach elsewhere [Chi00, Chi01]. We believe that, in this domain, approaches and studies should always be done using this user-centric approach.

6 Conclusion

As analysts, we are deeply involved in making the Web more accessible to users. We need to know what users are doing in order to better optimize the Web sites. Recent research seeks to understand the composition of user traffic using Web Mining techniques on server logs. The commonality is to first build up user profiles based on the user visitation paths, and then apply clustering techniques to these user profiles.

This paper summarizes several years of work. First, we presented a framework in which any of the available data features can be used in the clustering of user sessions. This framework enables robust combinations of content, linkage structure, and usage to discover user needs. Second, we presented results from a systematic evaluation of different clustering schemes by conducting a user study where we asked users to surf a large corporate site with *a priori* specified tasks. By knowing what the tasks were and how they should be grouped in advance, we were able to do post-hoc analysis of the effectiveness of different clustering schemes. We discovered that, by counting the number of correct categorizations, certain combinations of clustering schemes enabled us to obtain accuracies of up to 99%. While we don't necessarily expect this same level of accuracy on real world web logs, which are much noisier by nature, this is still quite encouraging to analysts hoping to make sense of user actions on the web.

We have since taken this knowledge and built LumberJack, a prototype automated analysis system that couples user session clustering with more traditional statistical analyses of web traffic. This combination seeks to not only identify significant user information goals, but also to understand how users follow these goals as they surf the site. The goal is to create a completely automatic system that quickly and accurately identifies a site's dominant usage patterns.

In the quest to understand the chain of user interactions on the Web, analysts need tools for getting quick and accurate pictures of Web usage. The work presented here suggests that in many cases the structure of user activity may be inferred automatically.

7 Acknowledgements

The authors would like to thank the comments and insights of George Karypis and Peter Pirolli. Pam Desmond helped with proofreading. This work was supported in part by Office of Naval Research grant No. N00014-96-C-0097 to Peter Pirolli and Stuart Card. Finally, we would like to thank the websites who have donated their web logs for analysis and all the subjects who participated in our evaluation.

References

[Banerjee01] Banerjee, A. and Ghosh, J. Clickstream Clustering using Weighted Longest Common Subsequences, in *Proc. of the Workshop on Web Mining, SIAM Conference on Data Mining* (Chicago IL, April 2001), 33-40.

[Barrett97] Barrett, R., Maglio, P.P., and Kellem, D.C. How to personalize the Web, in *Proc. of the ACM Conference on Human Factors in Computing Systems, CHI '97* (Atlanta GA, March 1997), 75-82.

[BenHur02] Ben-Hur, A., Elisseeff, A., and Guyon, I. A Stability Based Method for Discovering Structure in Clustered Data, in *Proceedings of the Pacific Symposium on Biocomputing (PSB2002)*, January 2002, Kaua'i, HI.

[Caruana94] Caruana, R., and Freitag, D. 1994. Greedy attribute selection. In *Proc.of International Conference on Machine Learning, ML-94*, pp. 28-36. Morgan Kaufmann.

[Chi00] Chi, Ed H., Pirolli, P., and Pitkow, J. The Scent of a Site: A System for Analyzing and Predicting Information Scent, Usage, and Usability of a Web Site. In *Proc. of ACM CHI 2000 Conference on Human Factors in Computing Systems*, pp. 161--168, 581, 582. ACM Press, 2000. Amsterdam, Netherlands.

[Chi01] Chi, E.H., Pirolli, P., Chen, K., and Pitkow, J. (2001). Using information scent to model user information needs and actions on the Web. *Proc. of the ACM Conference on Human Factors in Computing Systems, CHI 2001* (pp. 490-497), Seattle, WA.

[Cooley97] Cooley, R., Mobasher, B. and Srivastava, J. Web Mining: Information and Pattern Discovery on the World Wide Web. In *Proc. of the International Conference on Tools ith Artificial Ingelligence*, pp. 558-567. IEEE, 1997.

[CLUTO02] CLUTO: A Software Package for Clustering High-Dimensional Datasets. Available at http://www-users.cs.umn.edu/~karypis/cluto/

[Fu99] Fu, Y., Sandhu, K., Shih, M. Asho Generalization-Based Approach to Clustering of Web Usage Sessions, in *Proc. of WEBKDD 1999* (San Diego CA, August 1999), 21-38.

[Heer01] Heer, J. and Chi, E.H. Identification of Web User Traffic Composition using Multi-Modal Clustering and Information Scent, in *Proc. of the Workshop on Web Mining, SIAM Conference on Data Mining* (Chicago IL, April 2001), 51-58.

[Heer02a] Heer, J. and Chi, E.H. Mining the Structure of User Activity using Cluster Stability, in *Proceedings of the Workshop on Web Analytics, SIAM Conference on Data Mining* (Arlington VA, April 2002).

[Heer02b] Heer, J. and Chi, E.H. Separating the Swarm: Categorization Methods for User Access Sessions on the Web. In *Proc. of ACM CHI 2002 Conference on Human Factors in Computing Systems*, pp. 243--250. ACM Press, April 2002. Minneapolis, MN.

[Hong01] Hong, J.I., Heer, J., Waterson, S., and Landay, J.A. WebQuilt: A Proxy-based Approach to Remote Web Usability Testing, to appear in *ACM Transactions on Information Systems*.

[MacQueen67] MacQueen, J. Some methods for classification and analysis of multivariate observations, in Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, vol. 1 (1967), UC Berkeley Press, 281-297.

[Porter80] Porter, M.F., 1980, An algorithm for suffix stripping, *Program*, 14(3) :130-137

[Pirolli99a] Pirolli, P. and Pitkow, J.E. Distributions of Surfers' Paths Through the World Wide Web: Empirical Characterization. *World Wide Web*, 2(1-2), 1999. 29-45.

[Pirolli99b] Pirolli, P. and Card, S. K. (1999). Information Foraging. *Psychological Review* 106(4): 643-675.

[Schuetze99] Schuetze, H. and Manning, C. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge MA, 1999.

[Schuetze99b] Schuetze, Hinrich, Pirolli, Peter, Pitkow, James, Chen, Francine, Chi, Ed, Li, Jun. System and Method for clustering data objects in a collection. Xerox PARC UIR QCA Technical Report, 1999.

[Shahabi97] Shahabi, C., Zarkesh, A.M., Adibi, J., and Shah, V. Knowledge Discovery from User's Web-page Navigation, in *Proc. 7th IEEE Intl. Conf. On Research Issues in Data Engineering* (1997), 20-29.

[SIAM01] *Proc. of the Workshop on Web Mining, SIAM Conference on Data Mining* (Chicago IL, April 2001).

[Srivastava00] Srivastava, J., Cooley, R., and Deshpande, M. (2000) Web Usage Mining: Discovery and Application of Usage Patterns from Web Data. *SIGKDD Explorations* 1(2): 12-23.

[WEBKDD01] *Proc. of the SIGKDD Workshop on Web Data Mining (WEBKDD01)* (San Francisco CA, August 2001).

[Yan96] Yan, T.W., Jacobsen, M., Garcia-Molina, H., and Dayal, U. (1996), From User Access Patterns to Dynamic Hypertext Linking. *Computer Networks*, vol. 28, no. 7-11 (May 1996), 1007-1014.

[Zhao01] Zhao, Y. and Karypis, G. (2001). Criterion Functions for Document Clustering: Experiments and Analysis. Technical Report #01-40. University of Minnesota, Computer Science Department. Minneapolis, MN.