NISS

A Model for Relating Browser Behavior to Site Design on the World Wide Web

Murali Haran, Alan F. Karr,
and Ashish Sanil

Technical Report Number 159
September 2004

National Institute of Statistical Sciences
19 T. W. Alexander Drive
PO Box 14006
Research Triangle Park, NC 27709-4006
www.niss.org

A Model for Relating Browsing Behavior to Site Design on the World Wide Web

Murali Haran, Alan Karr and Ashish Sanil

1 Introduction

A critical problem for users of the World Wide Web is that many sites are difficult to navigate, hard to use and have confusing structure. Examples abound because the phenomenon is so widespread. Users may become lost, and they may make (or be forced to make) large leaps within a Web site (for example, returning unnecessarily to the home page) that are inconsistent with its structure. They may rely on their browser s back button as a navigational tool. They may be unable to find content and abandon the site. Obviously site authors do not frustrate their visitors intentionally. It is simply exceedingly difficult to create easy-to-use sites. A structure that seems intuitive to an author may be highly confusing to everyone else. Nor is there is any mechanism for site authors to understand at any but the most cursory level how visitors actually use the site. One way to improve usability is to conduct formal user studies and measure user performance for specific tasks, but this is too expensive for all but a few sites. A second approach is to exploit the rich instrumentation in the on-line world. This is the approach we adopt here.

Web servers create voluminous log files that record every hit to every page on the site. By processing the log files we can create sessions consisting of sequences of page views for each user. There are both commercial and public domain Web reporting tools that produce simple reports summarizing site activity. From these reports, however, it is essentially impossible to relate site activity to site structure, to correlate site modifications with activity changes, or to make stochastic predictions. To address this gap we propose to construct Bayesian statistical models that relate visitor transition patterns to site structure. The pages are grouped into more meaningful 'nodes', and the website is modeled as a tree, rooted at the home page, whose vertices are nodes. We apply the model to a major commerce website. Each user session is reduced to the entry page E, and a set of counts for each node corresponding to the number of each observed transition type originating from that node. Computation of the posterior distribution requires Markov chain Monte Carlo (MCMC) simulation). We find that grouping pages into nodes effectively deals with issues of scalability and ease of interpretability. This set of models is rich enough to support statistical inference about a number of important questions such as: Are user transitions consistent with the Web site structure? Are users more likely to go to "special pages" than take tree-consistent transitions?

In Section 2, we describe the nature of the weblog data from the commerce website, and in Section 3, we provide details of our model for the data, and explain how computation is performed efficiently for inference based on the model and data. We then describe the results of the application of these methods to the commerce data in Section 4, and conclude in Section 5 with a discussion of our results and directions for future work.

2 Data

The data consist of user sessions which are described as sequences of web pages, where each page has a unique identifier. Data was gathered for a period of 6 months in 2002. The number of visitors to the website during this time was 238,595, and the total number of sessions was 344,227. 5,575 distinct web pages appear in the data set. Note that we have removed all sequences of length 1, since a user who exits immediately after entering does not provide useful transition information (except, perhaps, to identify entry pages that lead to the most frequent immediate exits).

It is possible to produce a matrix of transition counts where the (i, j)th entry of the matrix is the number of observed transitions from page i to page j for all sessions involving all users. Since our goal is to study how users navigate the website, a natural first attempt at summarizing this information would be simply to divide the number of transitions in a particular cell by the total number of transitions for that row, and report this number as $TP(i, j)$, the estimated probability of moving from page i to page j. However, these probabilities fail to capture the structure of the website in any easily interpretable manner, especially when the number of pages is very large. It is also difficult for a website designer to describe overall design in terms of individual pages, particularly since individual pages may have been dynamically generated (there are potentially an infinite number of pages that can be dynamically generated.)

Instead, we use information from the web designer to provide a list of 316 nodes, which represent meaningful groupings of the 5575 pages. The design is then expressed as a tree containing all such nodes. The transitions that are consistent with the tree structure are parent-child (P), child-parent (C), self (R), and sibling (S). Several nodes are labeled as "special": Homepage(H), FAQ (F), Tutorial (U), Downloads (D), Fees (T), Images(I). A transition to a special node is also considered to be consistent with tree structure. Thus, by eliciting relevant information from the website designer, we now have a basis for assessing consistency of user behavior with the design of the website.

2.1 Data Reduction

We describe here how we process the data, as this is crucial to the formulation of our model:

1. Page sequence data: The raw data is in the form of several user sessions.Each session simply consists of a sequence of pages. Let a single session be: $p_1, p_2, \ldots, p_n$, where n is the length of the session. Since repeated pages are not indicative of user movement/transition through the website (and are often simply the result of an automated process of page 'refreshes'), we removed any such sequences of repeats.

2. Node sequence data: The information provided by the developers maps each of the pages in the data to a unique node, i.e. for each page, p, there is a corresponding node, ν. The design of the website is described in terms of these nodes; therefore, we translate each page into its corresponding node and obtain the sequence: $\nu_1, \nu_2, \ldots, \nu_n$.

3. Entry and exit nodes: We use an entry node, E that represents all pages outside the website from which the user first entered the website at node ν_1. Similarly, at the end of each session, we use an exit node, X to represent all destinations outside the website to which the user exits at the end of a session. The single session is now: $E, \nu_1, \nu_2, \ldots, \nu_n, X$. This is similar to the method used in Di Scala et al. (2003).

4. Page hierarchy information: we build the minimal tree that represents all nodes that appear in the data by using the page hierarchy information provided by the website designers. The tree is built in the following manner: for each node in the data, add the node and its parent node, if these nodes are not already in the tree, and add the corresponding link. Based on this tree, it is now possible to ascertain if the transition between nodes u and v represents a parent-child, child-parent, or sibling transition. A designed transition that does not qualify as one of these tree-transitions, may be either a self (repeat) transition or a transition to one of the special pages. A transition that does not fit into any of these categories is simply categorized as 'other'.

5. Transition matrix data: Using the page hierarchy information, we can classify a transition from any node to another node as one of 12 possible transitions: parent-child (P), child-parent (C),

self/repeat (R), sibling (S), to FAQ page (F), to homepage (H), exit (X), Tutorial (U), Downloads (D), Fees (T), Images(I), or to "other" (O). By counting the number of transitions of each type, it is possible to generate a transition matrix with rows corresponding to each of the nodes, and columns corresponding to the node-type transitions.

Note that there are some features of this problem, and the data processing that are unique:
(1) The node hierarchy information provides a clearly defined design associated with the website.
(2) The pages that appear in the data set are mapped to "nodes". This framework allows several closely related, and possibly dynamically generated pages, to belong to the same node. Therefore, by translating the page sequence data into node sequence (Step 2) above, user behavior on this website is more readily related to the organization of the website. Some advantages of this approach are:
(a) The designers can describe their design in a more meaningful and abstract fashion - they can avoid having to specify a possibly contrived tree-like structure that accounts for *all* pages. Instead, they can simply describe the structure of the website in terms of nodes, which are more intuitive and meaningful 'units' of the tree.
(b) We can greatly reduce the dimensionality of the problem, thereby improving scalability of any techniques used to model or study such data. Where a transition matrix summarizing the original pages without any page hierarchy information would have been of dimension $n_p \times n_p$(5575 × 5575 in our case), we can now reduce the number of columns to 12 node types, and the number of rows to n_t nodes, and therefore a $n_t \times n_p$ matrix (316 × 12 in our case). Such reductions may be even more dramatic and therefore even more crucial when dealing with scalability issues for larger websites, which have orders of magnitude larger number of pages. Since our primary goal is to understand the usage of the website in terms of the design of the website, we believe that a matrix of probabilities of nodes and their corresponding transition-types is a more meaningful summary of web usage than a large matrix of page to page transition probabilities.

It is important to note that the tree structure of the page hierarchy information serves as an intuitive means to describe the site design; however, in the way in which we use it here, the tree information essentially represents an efficient lookup table. For any pair of nodes (u, v), we see if it fits into any one of the tree-defined relationships: parent-child, child-parent, or sibling. If not, we look to see if $u = v$, or if v is one of the specially designated pages (such as FAQ, Homepage etc.)

3 Model

Denote the probability of a type j transition from some node i by $\pi_{i,j}$, and the transition probability from entry node to node i by $\pi_{E,i}$, where $i \in \mathcal{S}$. The likelihood for observing a particular sequence of nodes (as translated from the page sequence) is simply taken as the product of the probabilities of observing the transition-types associated with the sequence. For instance, for the sequence ν_1, ν_2, ν_3, if the transition from ν_1 to ν_2 and ν_2 to ν_3 were of type k and l respectively, the probability of observing the sequence would be $\pi_{1,k}\pi_{2,l}$.

Let $\mathcal{S}$ be the set of n_s (316) nodes that appear in the data. Let $\mathcal{T} = \{P, C, R, S, H, F, U, D, T, I, X, O\}$ be the set of n_t node types, and the subsets $\mathcal{D} = \{P, C, R, S\}$, the tree-based designed transitions, and $\mathcal{A} = \{H, F, U, D, T, I, X\}$, the designed transitions to special pages. The likelihood of the data, $\mathcal{U}$, given the parameters, Θ, is described as

$$L(\mathcal{U}|\Theta) = \prod_{i \in \mathcal{S}} \pi_{E,i}^{TC(E,i)} \prod_{i \in \mathcal{S}, j \in \mathcal{T}} \pi_{i,j}^{TC(i,j)} \qquad (1)$$

Denote $\pi_{E,i}$ for $i \in \mathcal{S}$ by π_E and $\pi_{i,j}$ for $j \in \mathcal{T}, i \in \mathcal{S}$ by π_i. We place a Dirichlet prior on the entry page probabilities, π_E.

$$\pi_E \sim Dir(0.5, \ldots, 0.5)$$

Similarly, we place a Dirichlet prior on each set of node-type transition probabilities, (π_i). We model these Dirichlet distributions as arising from a common set of parameters,

$$\pi_i \sim Dir(\alpha_P, \alpha_C, \alpha_S, \alpha_R, \ldots, \alpha_X, \alpha_O) \qquad (2)$$

where $\alpha_0 = \sum_{j \in \mathcal{T}} \alpha_j$. We complete the Bayesian formulation by placing a prior on the parameters of the Dirichlet distribution for the node-type transition probabilities. The prior distribution is:

$$p(\alpha_P, \ldots, \alpha_X) = |I(\alpha_P, \ldots, \alpha_X)|^{1/2}, \qquad (3)$$

where $|I(\alpha_P, \ldots, \alpha_X)|^{1/2}$ is Jeffrey's prior for Dirichlet parameters (Yang and Berger (1996)), which under certain conditions (cf. Kass and Wasserman (1996)) represents 'non-informativeness' over transformations of the parameters.

3.1 MCMC Computations

Due to the high dimensionality and intractability of the posterior (4120 dimensions), it is natural to use Markov chain Monte Carlo (MCMC) methods (Gelfand and Smith (1990), Tierney (1994)) for inference about the posterior distribution. For the entry page parameters, we can easily calculate the parameters of the Dirichlet posterior. For the parameters of the other posterior distributions, the MCMC algorithm alternates between the following two steps:
(i)Sample from the full conditional Dirichlet distributions of page transition probabilities. This is done for each page, i (one vector at a time).
(ii) Draw from the common parameters of the Dirichlet distribution (α_j's) by Metropolis-Hastings steps.

4 Results

As a consequence of our MCMC inference, we obtain posterior distributions for node transition probabilities, along with 12 parameters (αs) that represent the common part of their Dirichlet distributions. Many questions can be answered by using these samples, but we will limit ourselves to describing a few here; there is a related discussion in Subsection 4.1.

Since our primary goal is to understand how well the user behavior on the website matches the design of the site, we are most interested in learning about how likely the user is to follow the design of the website. The odds of taking a non-designed transition transition versus a designed transition from node ν_i is $\pi_{iO}/(1 - \pi_{iO})$. Using the samples, we can easily construct 95% credible interval for the posterior $\pi_{iO}|\mathcal{U}$; let the interval be (l, u). If $0.5 < (l, u)$, ν_i is a poorly designed node, and if $(l, u) < 0.5$, ν_i is a well designed node. Studying credible intervals over all nodes, we find that 12.6% of all nodes are poorly designed, while 42.4% of all nodes are well designed (there is not enough evidence to make either claim about the rest of the nodes.) Thus, our results suggest that, while the users are behaving in a manner that is often consistent with the design, the designers should redesign the website, targeting the poorly designed nodes.

Nodes that had the highest median value for π_{iO} are considered to be the worst nodes, and these tended to be fairly low in the node hierarchy (deep in the node tree), and generally had few children and many siblings. Using a separate model (which we do not describe here for the sake of brevity), we investigated the relationship between consistency with design for a node and the description of the node according to the web designers. We found that only the number of siblings had a significant (and negative) effect on how well user behavior for the node matched the design - perhaps a design and implementation where nodes have fewer siblings would help make the website more usable. The best nodes were typically related to downloads and simulations, which is not surprising, given that both these activities are fairly sequential.

We also found that for 85% of the nodes, $p\left(\sum_{k \in \mathcal{D}} \pi_{ik} > \sum_{k \in \mathcal{A}} \pi_{ik} | \mathcal{U}\right) > 0.9$, so special nodes appear to be more important destinations than tree-defined destinations. This high reliance on special pages suggests that users tend not to follow the tree design as much as the designers would like. On the other hand, since designed transitions include transitions to special pages, this may not necessarily be an undesirable result, particularly when the transitions are to special pages such as "Fees", "Tutorials" and "Simulations".

4.1 Advantages of our approach

Once we obtain the analytic form of the posterior distribution of the parameters, or are able to simulate from the posterior distribution, we can compute or estimate a number of relevant probabilities and other quantities of interest. We now describe how the effort required to work on a very general model under a fully Bayesian framework can yield significant rewards by making it possible to answer a number of important questions in a unified manner

Comparing Structure and Usage. We can better understand how the web site is being used by examining appropriate posterior probabilities, and we can use these as a guide for improving the Web site design. For example, the question “How likely are visitors not to respect the intended tree structure of the site?” can be answered by computing the probability, in light of the data, that a user will jump from any given node to one of the nodes that is not designated as a “child”, “parent” or “search”, “help” or “exit” node. Similarly, the question “Are users more likely to exit from node A or node B (to exit page, X) ?” can be answered by calculating $Pr(A, X > B, X|U)$, the probability that the user is more likely to exit from page A than from page B, and so on. A related example was presented in the previous section.

Comparisons and Trend. It is possible to use the posterior distribution to assess change in usage patterns between time periods, such as after a promotional campaign. For instance, a promotional campaign targeting content featured on page A could be evaluated as follows. First, obtain the posterior distribution A,E for page A how likely were users to enter the site from page A from data collected before the campaign was launched. Then, obtain the posterior distribution A,E for page A from data collected after the campaign was launched. Finally, we can compare the two distributions, either visually by plotting histograms, or more formally with tests such as the Kolmogorov Smirnov test. This strategy can be clearly applied to many other questions of interest, such as How do usage patterns change between weekends and weekdays? or How does usage vary for visitors who come from an on-line campaign from those who come from an off-line campaign? Importantly, we can not only detect but also quantify differences, potentially allowing cost-based evaluation of promotions.

Page Inter-relations. One benefit of the Bayesian models is that we can readily answer a number of detailed queries regarding dependence. For example, the question Are the transitions to parent from pages A and B correlated? can be answered by obtaining the joint posterior distribution of A,P and B,P from the overall posterior distribution and computing their correlation coefficient, or visualizing the joint distribution. Extending this line of reasoning, a sample of the parameters of the model from the posterior distribution can be thought of as another (but richer) data set. It would then be possible to run data-mining tools [31] to discover interesting aspects of site usage such as associations between user behavior at various pages. The advantage is similar to bootstrapping: by means of resampling, faint effects are amplified to the point of detectability.

Simulation and Prediction. The posterior distribution of the parameters of this probabilistic model can be used to simulate realistic user sessions in order, for example, to test hardware or server capabilities. This differs from the usual simulation scenario where the distribution simulated from is taken to be known. In our case, the simulation requires two steps: The first samples a set of parameters from the posterior distribution, which are then used to generate a user session using the likelihood. An important advantage of this approach is that it enables us to generate a realistic set of user sessions that incorporates the uncertainty associated with estimating unknown parameters. From another perspective, this simulation procedure can be thought of as based on the model s predictions and thus it subsumes the issue of predicting user behavior. Such predictions can be applied, for example, to forecast user demand or the economic demand of a commercial campaign.

5 Discussion

We have developed a method that uses expert information to easily reduce the dimensionality of the problem. The designers are more easily able to describe their notion of the design of the website, and their concept of desirable user behavior. By reducing the dimensionality of the resulting transition probability matrix, we are able to also increase interpretability; we believe this is even more vital as people begin to study usability for much larger websites.

For the commerce data we have analyzed here, it appears that browsing behavior does not completely follow the design of the site (as narrowly defined by the node hierarchy and special pages). However, we believe that there are still several issues to be resolved regarding the description of the website. In particular, it is not always clear that a tree-like structure is adequate, given the increasing complexity of web designs. There are too many common links appearing on multiple pages, which makes it difficult to accomodate them all as 'special nodes'. It is also possible that other characterizations of the web designers' concept of poor usability would lead to fairly different assessments of the website. For instance, if it were possible to clearly define the notion of a 'lost' user, for eg. someone with frequent transitions to homepage and exit pages, we could associate high occurrence of lost users with poor design. Another possibility would be to try to classify whether a particular user session was successful - many failures would then imply poor design. However, while this is an attractive notion for e-commerce, it may be difficult to assess success for other kinds of web designs.

There are several general options for the web designer to make use of the kind of analysis presented here. Nodes that are designated as particularly important origins can be modified in useful fashions to take advantage of the high traffic. For instance, more ads can be added to pages belonging to that node, and other important web pages can be linked from there as well. A reverse scenario may also be observed: parent-child transitions that have very low probability of being used suggest that the developers need to make sure that a prominent, easily-used link exists from the parent to the child node, or that they need to rethink their design and figure out if that parent-child transition truly exists. Finally, special attention needs to be paid to nodes that appear to be important destinations, but to which transitions do not follow any of the standard transition types - perhaps this represents a gap between their design and the actual use of the website.

References

Di Scala, L., La Rocca, L., and Consonni, G. (2003). A bayesian hierarchical model for the evaluation of a web site. *Technical Report*, page NA.

Gelfand, A. E. and Smith, A. F. M. (1990). Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association*, 85:398–409.

Kass, R. E. and Wasserman, L. (1996). The selection of prior distributions by formal rules (Corr: 1998V93 p412). *Journal of the American Statistical Association*, 91:1343–1370.

Tierney, L. (1994). Markov chains for exploring posterior distributions (Disc: p1728-1762). *The Annals of Statistics*, 22:1701–1728.

Yang, R. and Berger, J. (1996). A catalog of noninformative priors. *Technical Report*, page NA.