Robust Models of Mouse Movement on Dynamic Web Search Results Pages

Fernando Diaz*
Microsoft Research
fdiaz@microsoft.com

Ryen W. White
Microsoft Research
ryenw@microsoft.com

Georg Buscher
Microsoft Bing
georgbu@microsoft.com

Dan Liebling
Microsoft Research
danl@microsoft.com

ABSTRACT

Understanding how users examine result pages across a broad range of information needs is critical for search engine design. Cursor movements can be used to estimate visual attention on search engine results page (SERP) components, including traditional snippets, aggregated results, and advertisements. However, these signals can only be leveraged for SERPs where cursor tracking was enabled, limiting their utility for informing the design of new SERPs. In this work, we develop robust, log-based mouse movement models capable of estimating searcher attention on novel SERP arrangements. These models can help improve SERP design by anticipating searchers' engagement patterns given a proposed arrangement. We demonstrate the efficacy of our method using a large set of mouse-tracking data collected from two independent commercial search engines.

Categories and Subject Descriptors

H.1.2 [**Information Systems**]: User/Machine Systems

Keywords

Cascade model; Mouse-tracking

1. INTRODUCTION

Understanding how a user reads a search engine results page (SERP) is a fundamental concept in information retrieval. Linear scanning of a ranked list of documents underlies almost all classic evaluation metrics [35, Chapter 7] and, as a result, many of the formal models of information retrieval [31]. With access to large amounts of user click data, production web search engines have refined classic evaluation metrics [7] and retrieval models [23]. Nevertheless, the core assumption remains the linear scanning of a ranked list.

The linear scanning assumption may be inappropriate for many search services, including portal web search engines. These systems provide the user with SERPs which include much more than a ranked list of documents. Consider the example SERP in Figure 1. A SERP will include varying numbers of advertisements [5], query suggestions [28], and media-rich vertical content [1]. There is support for the claim that increasing the number of modules on the page can affect task completion [34]. Furthermore, the diversity of modules in an interface can also impact user experience and scan order [24, 27].

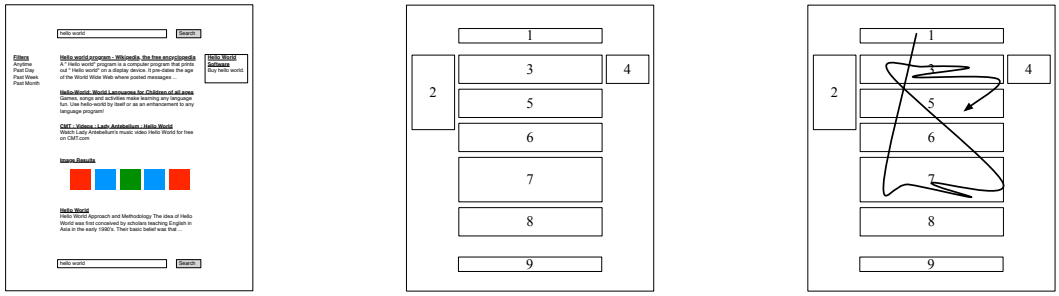
Modeling how a user reads a SERP is complicated by the fact that, for any two queries, the number and variety of modules on the page may be different. Queries suspected of having 'local intent' may be served SERPs which include a map. Queries suspected of having 'image intent' may be served inline images. We expect, in practice, to observe a large diversity of arrangements. Figure 2 presents the distribution of unique arrangements in production search traffic for two different search engines (data collection is described in Section 6.1). The most frequent arrangement accounts for 8% of the total impressions in Search Engine 1 and 3% in Search Engine 2. The top 10% most frequent arrangements account for 82% of the total impressions in Search Engine 1 and 91% in Search Engine 2. On the other hand, 65% of arrangements occur only once in Search Engine 1 and 68% in Search Engine 2. That is, the majority of arrangements have very limited log data.

We propose a user model which generalizes linear scanning and incorporates ancillary page modules. Figure 3(a) depicts the classic linear scan model in graphical form. Labeled nodes represent page modules where m_1-m_5 represent documents, and m_0 and m_6 represent ancillary modules such as advertisements or query suggestions. The unlabeled nodes represent the start and end of a session. The edges represent the probability of transitioning between nodes. Much of the recent work on click modeling focuses on estimating the values of these edges [10]. This figure conveys two omissions in the linear scan model: ancillary modules and nonmonotonic transitions. Figure 3(b) depicts our generalization of the scan model. We consider the same modules but introduce edges between all nodes. As with click modeling, we will focus on estimating the values of these edges.

*Work partially conducted at Yahoo Research New York.

CIKM'13, Oct. 27–Nov. 1, 2013, San Francisco, CA, USA.

Enter the DOI string/url from the ACM e-form confirmation ...$15.00.



(a) Presentation (b) Arrangement (c) Mouse Data

Figure 1: Module-level representation of mouse-tracking data. The session sequence for this data would be $[1, 3, 5, 6, 7, 6, 5, 3, 5]$.

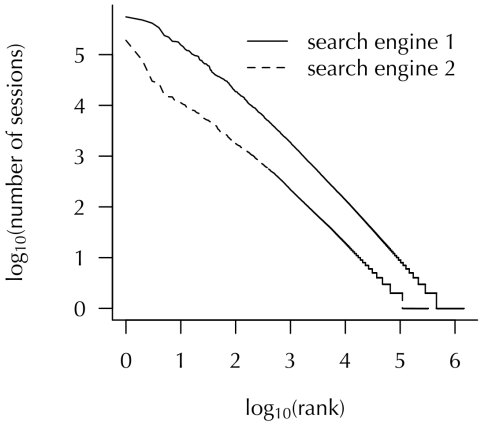
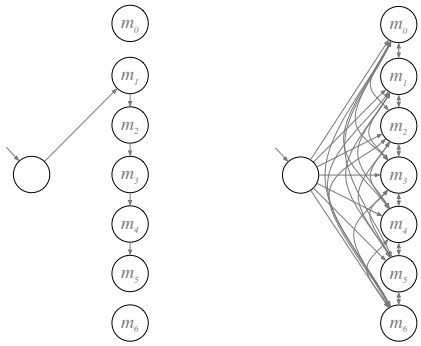


Figure 2: Distribution of unique page arrangements for SERPs from two large scale web search engines. The horizontal axis indicates the rank of the arrangement when sorted by frequency. The vertical axis indicates the frequency of that arrangement.



(a) linear (b) relaxed

Figure 3: The linear scan model and its relaxation.

In addition, we propose a user model which allows us to generalize to arbitrary page arrangements. This is important because previous user models based on click logs all assume a single topology across all queries. That is, by ignoring non-web modules, the graph structure in Figure 3(a) is shared across all queries. In our case, the topology in Figure 3(b) might be different for two arbitrary queries. Therefore, the edge weights learned for one query will be useless of a novel arrangement (topology).

In order to estimate the parameters of our user model, we exploit user mouse behavior associated with a SERP arrangement. We adopt this strategy because of the high correlation in general between eye fixation and mouse position [9]. Previous work has confirmed this correlation for SERPs [32, 17].

The focus of our study will be on the problem of constructing robust models able to make predictions about mouse behavior on arrangements for which we have little or no data available. Having such models provide a tool which can be used when manually designing new pages [33]. At a larger scale, mouse-tracking models could be useful for retrospectively detecting 'good abandonments', cases where the user was satisfied without clicking a link [22].

In this paper, we make the following contributions,

- a generalization of the linear scan model.
- an efficient and effective method for estimating the generalized model.
- an efficient and effective method for estimating parameters of unobserved arrangements (topologies).
- experiments reproduced on data sets from two large commercial search engines.

2. RELATED WORK

The motivation for capturing mouse movement at scale originates from results demonstrating a strong correlation between eye and mouse position [9]. In the context of web search, this correlation has been reproduced on SERPs [15], suggesting that, with some care [17], we can use logged mouse data as a 'big data' complement to eye-tracking studies [3]. Such studies have found that mouse-tracking is useful for click prediction [18] and advertisement interest prediction [14]. In fact, mouse movement analysis has been suggested as useful for web site usability analysis in general [2, 3]. Even without assuming a relationship between eye and mouse, important search signals such as query intent [13] and document relevance [19] can be detected.

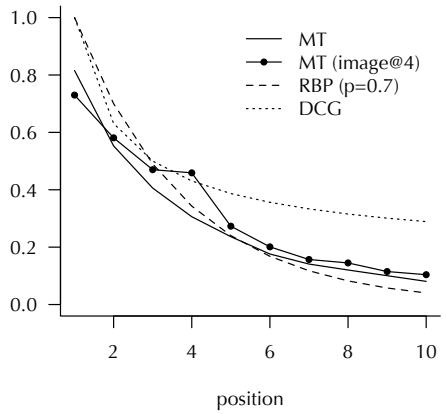


Figure 4: Position discount weights compared to examination inferred from mouse-tracking data for two layouts.

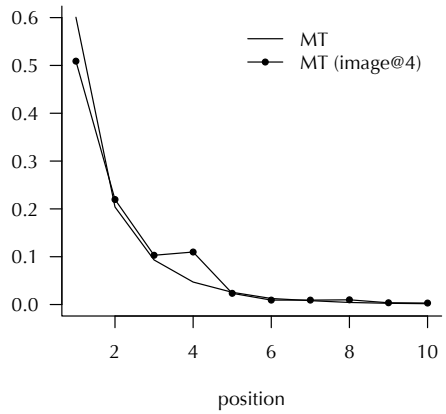


Figure 5: Probability of starting at different rank positions based on mouse-tracking data.

There has been some work on relaxing the linear scan user model. Wang *et al.* propose the application of partially observable Markov (POM) models to address non-sequential user models [37, 16]. This work is conceptually very close to our own with one important difference. The POM model assumes a fixed topology and cannot generalize to novel page arrangements (topologies). Punera and Meguru present a method for modeling nonmonotonic scan data but limit analysis to click data with an underlying ranking [30].

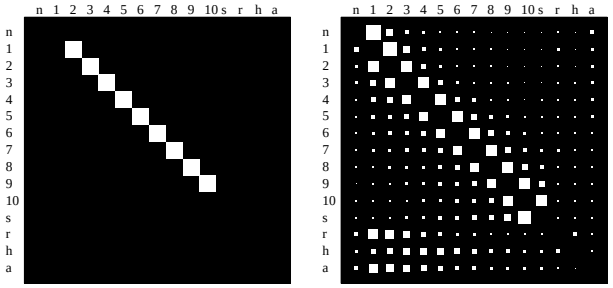
Modeling searcher behavior on SERPs is a fundamental part of the feedback used in web search engine design. Logging simple click and skip statistics can be exploited to improve ranking performance [20]. More sophisticated models of user interaction also based on click information include assumptions about user satisfaction [10], result attractiveness [8], and document utility [11]. Since most existing SERP models leverage data only from user click information, our work can be seen as an extension of these models to incorporate mousing data.

Outside of web search, there exist many models of visual attention. Models include those based on the visual salience [29], and biology [4]. Our work is most closely related to machine learning models of visual attention [6, 21]. In these cases, the authors attempt to predict the eye-tracking data from a small eye-tracking study using image-based signals. Practical issues prevent these experiments from being conducted on larger populations. These issues include the expense of storing heavy image data for each SERP and reliably transferring and/or rendering a user's precise layout. Our work can be seen as an efficient extension of these models to large data sets with mouse-tracking data. The addition of such data provides a more complete representation of search activity, which may be useful in developing more accurate behavior models.

3. MOTIVATING ANALYSIS

Several offline and online evaluation metrics make assumptions about the relationship between document rank position and examination. In this section, we will investigate the support for these assumptions in mouse-tracking data.

The models underlying metrics such as discounted cumulative gain (DCG) and rank biased precision (RBP) assume that the probability of examination of an item is conditioned only on its position. In Figure 4, we compare position discount weights based on DCG and RBP to probability of examination based on mouse-tracking data for two layouts, a standard '10 blue links' layout and a layout including an image vertical at position 4. We consider a module examined if the user moused over it during a page view. We then normalize, per position, by the number of page views, considering page views which include a mouse-over on at least one result. There are two noteworthy findings from this plot. First, neither of the two empirically-derived probabilities observe probability 1 at the first position. This implies that, for roughly 20% of the page views, users never moused over the first result, even though they moused over others. Second, page views with an image have significantly different probabilities of examination for each position compared to the 'standard' SERP. Compared to the position models, the empirical probabilities are flatter, suggesting more examination than is suggested by those models. We will explore why this is later in this section.



(a) Linear Scan (b) Actual Mousing

Figure 6: Hinton diagrams representing the probability of transitioning between pairs page modules, including the ten algorithmic results as well as navigational modules (n), query suggestion (s), related searches (r), search history (h), and advertisements (a). Each row shows the conditional distribution of the second position (column id) given the first position (row id). Figure 6(a) reflects the transition probabilities assumed under the linear scan assumption. Figure 6(b) displaying the empirical transition probabilities from mouse-tracking data for pages including only 'ten blue links'.

The linear scan model makes three assumptions about user behavior: the user begins at the top of the ranked list, the user reads in order, and the user only interacts with the algorithmic results. In order to test the assumption that the user begins scanning at the top of the ranked list, for each module, we measured the fraction of impressions for which it was the first module moused over. If Figure 3(a) is accurate, then only the first module should receive any mass. In Figure 5 we show that, while many impressions begin at the top of the list, roughly 40% begin at another position for the standard SERP and 50% for the SERP with an image result. The drop in first position probability can be accounted for by the increased probability of users immediately mousing over the image result at the fourth position. This is consistent with previous studies demonstrating image attractiveness [25, 36]. We tested the assumption that users read in order by computing the probability of transitioning between pairs of items. In Figure 6, we present Hinton diagrams for the top 6 results for the linear scan model (6(a)) and a model derived from mouse-tracking data for a SERP with ten standard search results (6(b)). We observe that, although users do have a tendency to mouse downward, there is significant movement skipping results and moving backward. Again, this is consistent with previous studies [25]. We also notice that, although the majority of transitioning appears to occur in the web results, there is non-negligible attention on ancillary modules. Recent studies suggest that this is more pronounced in right panels containing entity information [26].

We can also compare the deviation in transition probabilities between pages with standard ordering and those with vertical content. Figure 7 compares the transition probabilities between pages with standard ordering and those with vertical content. We focus our analysis on the first six positions to make the matrix dimensions comparable. When weather vertical content occurs at position 1, we notice that there is a significant rise in the probability of transitioning to position 1 (Figure 7(a)) and a significant drop in the probability of transitioning to positions 3-6 (Figure 7(b)). When image vertical content is presented at position 4, we observe an increase in engagement with position 4 (Figure 7(c)) and a corresponding drop in transitions to positions 5 and 6 and from 5 and 6 to 2 and 3 (Figure 7(d)).

We also note that, although we can reliably compare mouse behavior on the first six positions (Figure 7), doing so for the entire matrices of two arrangements is problematic. The fundamental problem is that the dimension of these matrices will depend on how many modules are present in the arrangement. For example, both of the arrangements with vertical content technically consist of 11 positions in the center of the page whereas the reference arrangement consists of 10 positions. Even if the number of modules is identical for two arrangements, the semantics for spatial position of the module may be different between two arrangements.

The analysis in this section suggests that actual user scanning behavior deviates significantly from existing user scanning models. The core inconsistency results from (a) inconsistent starting position, (b) nonmonotonic scan order, and (c) presentation bias. In the following sections, we will describe an experimental framework for designing and evaluating models which address these problems.

4. PROBLEM DEFINITION

We are interested in modeling the mouse movement between significant regions of the SERP. Accurate models of mouse movement facilitate the predictive analysis of searcher attention with rare or unseen SERPs. In our work, significant regions include non-overlapping bounding boxes of individual search results, advertisements, logos, query suggestions, and navigation modules (Figure 1). We refer to these boxes as *modules*. For each type of module, there are a large variety of factors which may affect user behavior. For example, search results may consist of text only or include an image; a vertical result may be innocuous as with a calculator computation or large and media-heavy as with an important news display.

An *arrangement* of modules refers to the relative positioning of modules on a page. For example, 'ten standard web results and one advertisement on the right' would be one arrangement; 'ten standard web results and one advertisement on the top' would be a second arrangement. An arrangement is the result of decisions from several page layout algorithms (e.g. vertical selection, advertisement auctions). Note that, because a given SERP can only contain a fixed number of modules, not every type of module will be represented in an arrangement. Furthermore, some module types (e.g. the web search result type), will be represented by several modules in an arrangement. Let $\mathcal{M}$ be the set of modules for a given arrangement presented to the user. The user's *session sequence* is defined as,

$$\mathbf{s} = [s_1, s_2, \ldots, s_t]$$

where $s_i \in \mathcal{M}$ indicates the ith module visited. Our data set, $\mathcal{D}$, consists of a set of session sequences, one for each query submitted to the search engine by an individual user at a specific time. A sequence can be considered a trace through the Markov model depicted in Figure 3(b).

Our task is to, given a subset of $\mathcal{D}$ used for estimating a model (the training data), make predictions on a separate *testing* set of sequences. Our model will be probabilistic. Specifically, given $\mathcal{M}$, we are interested in estimating the probability of transitioning between all pairs of modules. If $n = |\mathcal{M}|$, then we represent these probabilities using the $n \times n$ stochastic matrix $\mathbf{P}$. Each row represents a transition from a module i to all other modules; therefore,

$$\sum_{j=1}^{n} P_{i,j} = 1 \tag{1}$$

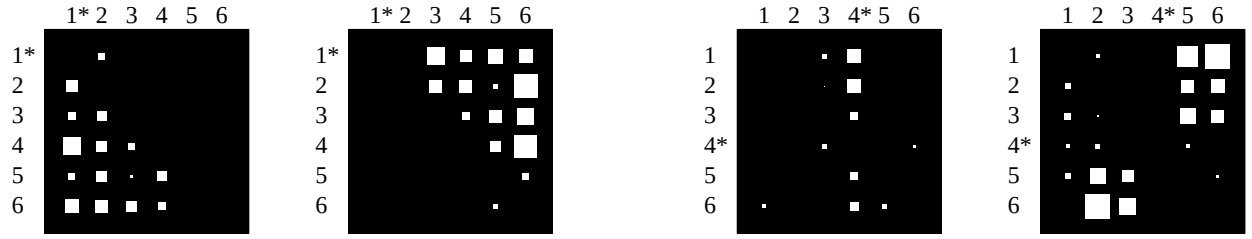
Such a model is also known as a first-order Markov chain.

We will be evaluating two types of predictions. Because it is probabilistic, our model can be evaluated according to the likelihood of it having produced the test set. Specifically, the likelihood of producing an individual session sequence is,

$$\mathcal{L}(\mathbf{s}|\mathbf{P}) = \prod_{i=1}^{t-1} P_{s_i, s_{i+1}} \tag{2}$$

In addition we use a second, rank-based metric. Let $\rho(s_i)$ be the ranking of all modules given that we are in module s_i. Further, let $\rho(s_j|s_i)$ be the rank of module s_j given that we are in context s_i. We define the *mean reciprocal rank* of the session $\mathbf{s}$ as,

$$\text{MRR}(\mathbf{s}) = \frac{1}{|\mathbf{s}| - 1} \sum_{i=1}^{|\mathbf{s}|-1} \frac{1}{\rho(s_{i+1}|s_i)}$$



(a) $P^{\text{wea}}_{i,j}/P_{i,j} > 1$ (b) $P_{i,j}/P^{\text{wea}}_{i,j} > 1$ (c) $P^{\text{img}}_{i,j}/P_{i,j} > 1$ (d) $P_{i,j}/P^{\text{img}}_{i,j} > 1$

Figure 7: Ratios of transition probabilities between pages with standard ordering and those with vertical content. Figures 7(a) and 7(b) consider unclickable weather content in position 1. Figure 7(a) depicts the ratio of transition probabilities where the weather layout results in higher transition probabilities; Figure 7(b) depicts the ratio of transition probabilities where the standard layout results in higher transition probabilities. Figures 7(c) and 7(d) present the similar plots considering an image vertical display at position 4.

If a system always ranks the observed module highest, then its performance will be 1. This second metric can be interpreted as evaluating a model ordering of modules as opposed to the quality of the probability estimates.

5. ALGORITHMS

Under the constraint of Equation 1, we will define several methods of estimating $\mathbf{P}$ given a training set $\mathcal{D}_{\text{train}}$.

5.1 Maximum Likelihood Model

Our first model selects the matrix $\mathbf{P}$ which has the maximum likelihood given $\mathcal{D}_{\text{train}}$. The values of $\mathbf{P}$ maximizing Equation 2 on $\mathcal{D}_{\text{train}}$, are

$$P_{i,j} = \frac{D_{i,j}}{\sum_k D_{i,k}} \tag{3}$$

where $\mathbf{D}$ is an $n \times n$ matrix of counts of the number of transitions from i to j observed in the data set $\mathcal{D}_{\text{train}}$.

Even though the maximum likelihood model, as one might suspect, is the model which optimizes the evaluation metric on a particular data set, it is not necessarily the best model for the test set. For example, suppose we observe the transition (i, j) in the testing set. If this transition was not observed in the training set, then $P_{i,j} = 0$ which, in terms of likelihood, is the worst estimate one could compute. In fact, if we have not made any observations of the mouse exiting module i, then Equation 3 is undefined. Relatedly, sometimes we have prior information about the probability of a transition. For example, we may know that a user is likely to transition from i to j if i is above j; less so if they were reversed. There is no elegant method of incorporating this knowledge into the model. Perhaps the biggest shortcoming of the maximum likelihood model is the inability to generalize to novel page arrangements. This is because the elements of $\mathbf{P}$ have very rigid interpretations. Suppose module i represents a standard web snippet at the top position. If we are presented with an arrangement which replaces that result at the top position with an image or advertisement, then we cannot reliably claim the transition probabilities are consistent. Nevertheless, if the training data is plentiful and representative of the testing data, then the probability estimates will be reliable.

5.2 Farley-Ring Model

feature type	features
module	w, h, $w \times h$, x, y, DOM class
interaction	same id, same class, left of, above, min distance, area ratio
page	w, h, $w \times h$, num. modules, time of day
viewport	w, h, $w \times h$

Table 1: Features used by the Farley-Ring model to estimate the probability of transitioning between modules i and j. Each feature type is represented once except for module features which are defined for each i and j under consideration. DOM class refers to the type of content in the module (e.g. web result, image, advertisement).

In order to address some of the limitations with the maximum likelihood model, we propose a second model which does not require any observed mouse data on a particular arrangement in order to make predictions. We can accomplish this by learning the relationship between properties of modules and the probability of transitioning. Recall our example of prior knowledge. We would like to learn the relationship between 'i is above j' and the probability of the mouse moving from i to j. In fact, for two modules i and j, we can make a large set of statements about i and j (e.g. 'i contains an image', 'j is an advertisement'). We would like to learn the relationship between this set of statements and the probability of the mouse transitioning from i to j.

More concretely, we treat the task of modeling $\mathbf{P}$ as a regression problem. To this end, we treat the statements about i and j as *features* and the transition from i to j as the *target*. Let $\phi_{i,j}$ be a vector of the values of these variables for a particular pair (i, j). We list the complete set of features in Table 1.

It is worth describing how we learn this relationship. Assume that, for a particular session with n modules, we observed the user mousing *into* module i. If the mouse transitioned from i to j, we treat an observed movement from i to j as having target $+1$ and transitions to all other modules (except i) as having target -1. Performing this operation on all of the sessions in $\mathcal{D}_{\text{train}}$ results in a large set of data $\mathcal{D}^{\phi}_{\text{train}}$.

Having explained the representation of data used to model $\mathbf{P}$ with pairwise features, we can now describe how we train

our model. Because $\mathcal{D}^{\phi}_{\text{train}}$ represents independent variables with a binary target, we adopt a logistic regression model which we parameterize by a vector of coefficients, β. These coefficients can be used to predict a probability of transition given a pair of modules i and j, $f(i, j; \beta)$. Although the range of f is the interval $[0, 1]$, Equation 1 constrains $\mathbf{P}$ to be row normalized. Therefore, our final model is defined as,

$$P_{i,j} = \frac{f(i, j; \beta)}{\sum_k f(i, k; \beta)} \tag{4}$$

The logistic regression model is robust insofar as it is able to generalize to arbitrary page layouts, so long as ϕ can be computed. Fortunately, the features in Table 1 are simple properties that can be computed for unseen page arrangements and even new modules. Farley and Ring developed a similar approach in the context of modeling supermarket traffic flow for arbitrary aisle arrangements [12].

Although this model is generic to the extent that it uses arrangement-agnostic features, the actual portability of the model depends critically on $\mathcal{D}_{\text{train}}$. If the training data is taken exclusively from a single module arrangement, we can imagine that it will not learn the relationship between certain underrepresented features and transitions. We will explore the sensitivity of the model on $\mathcal{D}_{\text{train}}$ in Section 7.

5.3 Updating the Farley-Ring Model

In some cases, we have access to small amount of training data on an arrangement of interest, perhaps from a small eye-tracking study. Although we can use this data directly with the maximum likelihood model, it is unclear precisely how to incorporate it into the Farley-Ring model.

One way of adding data to the Farley-Ring model is to use *Bayesian updating*. In order to do this, we model $\mathbf{P}$ as a random matrix. That is, we will assume that the elements of $\mathbf{P}$ are drawn from some distribution. Recall that each row, $\mathbf{P}_i$, defines a multinomial (Equation 1). Therefore, we need a distribution over multinomials. We adopt the Dirichlet distribution whose probability density function is defined as,

$$g(\mathbf{P}_i; \alpha_i) = \frac{\prod_{j=1}^{n} \Gamma(\alpha_{i,j})}{\Gamma(\sum_{j=1}^{n} \alpha_{i,j})} \prod_{j=1}^{n} P_{i,j}^{\alpha_{i,j}-1} \tag{5}$$

where $\alpha_{i,j} = \mu \frac{f(i,j;\beta)}{\sum_k f(i,k;\beta)}$ and μ is a free parameter. We define such a Dirichlet distribution over each row of the matrix, sharing the same value for the parameter μ. Given data on an example arrangement, because the Dirichlet distribution is the conjugate prior to the multinomial distribution, the posterior distribution is also a Dirichlet. Since we evaluate our models according to a fixed estimate of $\mathbf{P}$, we use the posterior means of the multiple Dirichlet distributions to evaluate our model. Fortunately, the posterior mean, given $\mathcal{D}_{\text{train}}$, can be computed in closed form,

$$\hat{P}_{i,j} = \frac{D_{i,j} + \alpha_{i,j}}{\sum_k D_{i,k} + \mu} \tag{6}$$

It is worth pointing a few things out in this equation. First, notice that, if there is no training data, then $\mathbf{D}$ contains only zeroes and Equation 6 is equal to Equation 4. Second, as data sets grow very large, we quickly approach Equation 3 because the values in $\mathbf{D}$ will dominate $\alpha_{i,j}$ and μ. Finally, the parameter μ reflects our confidence in the original model. For large values of μ, we trust the prediction from the original model and require proportionally more example data to adjust our estimates.

6. METHODS AND MATERIALS

6.1 Data Collection

The set of relevant modules includes search results, vertical results, advertisements, logos, navigational menus, and search boxes. We collected our data on two separate large scale commercial search engines using the same methodology. For a small set of search users, we instrumented the HTML content of the SERP with JavaScript to detect mouse-overs on relevant modules. For each query, we recorded the event of a mouse entering or leaving a module on a SERP (Figure 1). For Search Engine 1, we conducted this experiment for two days of production traffic in late summer 2011. We observed a total of 324,235 unique arrangements and 2,356,907 sessions. For Search Engine 2, we conducted this experiment for several days of production traffic in spring 2012. We observed a total of 1,454,256 unique arrangements and 19,874,523 sessions. No personal data besides the order of modules visited was used in our experiments.

6.2 Sampling Training Data

As mentioned in Section 5.2, the generalizability of the Farley-Ring model may be sensitive to the training data. In order to measure the sensitivity, we consider three different models, each trained using a different sampling strategy for training data. Our first training strategy, referred to as 'top', uses a large number of sessions from the most frequently occurring arrangement. We hypothesize that this approach will not capture arrangement-agnostic relationships between features and transitions. We propose two alternatives to this baseline. First, we can sample sessions randomly from our data set (i.e. the distribution in Figure 2); we refer to this run as 'random'. We expect that this training set will be more representative of the diversity of module arrangements encountered in production. Second, we can sample sessions so that each arrangement has equal representation in the training set; we refer to this run as 'round-robin'. We expect this training set to explicitly attempt to represent the diversity of module arrangements. All training sets, regardless of sampling strategy, contained the same number of sessions.

6.3 Experiments

We consider three experimental setups. In the first setup, we test the behavior of our algorithm in the presence of a single target arrangement. We take the most common arrangement and hold 20% of the sessions for evaluation and vary the training conditions according to Section 6.2. In the second experiment, we test the robustness of our algorithm in making predictions for a diverse set of target arrangements. We accomplish this by evaluating our algorithms on sessions with the least common arrangements. The exact number of evaluation sessions is equal to the number of evaluation sessions in the first experiment. In the third experiment, we test the Bayesian updating of Farley-Ring models to novel layouts. In order to evaluate the performance as a function of the amount of target arrangement data, we selected 25 arrangements with 1000 or more sessions. For each arrangement, we use 30% of the sessions for evaluation

and the remaining data for updating our Farley-Ring models according to Equation 6. In no cases are sessions from the evaluation set included in any of the models' training set. Free parameters are tuned on a separate validation set.

7. RESULTS

7.1 Base Experiments

The purpose of our first experiment was to test the situation where we are interested in a single arrangement, the most frequent arrangement. The results of this experiment are presented in Table 2. Expectedly, the model whose training data used sessions from the top arrangement performed best. Of the two sampling strategies, random sampling performed better, perhaps because it sampled top arrangements with more frequency than round robin sampling.

The purpose of the second experiment was to test the situation where we are interested in arbitrary novel arrangements with no observed data. The results of this experiment are presented in Table 2. Notice that the order of performance reverses compared to our first experiment. The model using training data from a single arrangement (top) fails to generalize to these novel arrangements because it has observed very biased examples of user behavior. On the other hand, our model which sampled training data to diversify page arrangements performs well because it has learned a generic, portable model. The only exception in this set of experiments is in the performance for the reciprocal rank metric with Search Engine 2 where random sampling outperforms round robin sampling.

The purpose of the third experiment was to test the situation where we had access to some training data on the evaluation arrangement. The results of this experiment are presented in Figure 8. We varied the number of sessions used to update the model. As we found in our second experiment, for little or no training data, our model based on round robin data outperforms the model based on data from a single arrangement. We also plot the performance of the maximum likelihood model (Equation 3). This model can be thought of as the performance if we did not invest the effort into training a Farley-Ring model. As we mentioned in Section 5.1, we can observe the maximum likelihood model performing well with many sessions. However, when we have few enough sessions (200 sessions for Search Engine 1, 60 sessions for Search Engine 2), using the updated Farley-Ring models performs significantly better than the maximum likelihood model. We also present the value of μ selected during validation. As we accumulate more observations of the target arrangement, the algorithm learns to automatically reduce the value of μ and become more similar to the maximum likelihood model.

7.2 Detecting Attention Deviation

One of the immediate advantages of a good estimate of $\mathbf{P}$ is the ability to visualize user behavior. Because our model is a Markov chain, we can examine its statistical properties to determine module importance. The stationary distribution of a Markov chain refers to the distribution over $\mathcal{M}$ representing the probability of being in a module as the length of the sequence goes to infinity. The stationary distribution can be computed by taking powers of the transition matrix, $\pi = \mathbf{P}^k\mathbf{e}$, for a large value of k. Equivalently, we can compute the left eigenvector of $\mathbf{P}$. The stationary distribution will capture position and presentation biases. However, it will also capture less interesting insights such as transitions based purely on geometry (i.e. two modules being adjacent). In order to normalize for this, we can compute the stationary distribution of an unweighted planar graph based on the adjacency of modules. That is, an edge has weight 1 if two modules are immediately adjacent, and 0 otherwise. Let π_{planar} be the stationary distribution of the unweighted planar graph. Our metric of module importance is defined by, $\pi - \pi_{\text{planar}}$. We should be clear that this metric, while well-motivated, relies on the unrealistic model of a user infinitely examining a SERP. We computed this metric for the arrangement consisting of standard web search results with an image vertical at position 4 (Figure 4). The results are presented in Table 3. The data suggests that the image vertical captures notably more attention than is expected from a purely geometric, uninformed model. Interestingly, the purely geometric model overestimates the importance of ancillary modules.

7.3 Extrapolating to Novel Arrangements

While the three experiments demonstrate the ability of the Farley-Ring model to make predictions on rearrangements of standard modules, we can test the ability of the model to extrapolate by using artificially-created arrangements. In Figure 9, we present the model predictions for various grid layouts of modules. We stress that these arrangements and module shapes were never observed in any of the training instances. Unlike previous work, our model provides a unique ability to extrapolate to completely novel presentations. In Figure 9(a), the arrangement of 25 web result modules is predicted to have an initial visitation concentrated in the top left, consistent with our intuition of user scanning behavior. If we replace one of the web results with an image result (9(b)), we observe the probability significantly increasing for that module. Nevertheless, a muted top left bias persists. If we replace four of the web results with a large web result (9(c)), we also observe it gathers more attention, though the impact on top left modules are less impacted. Finally, if we introduce a large image (9(d)), it receives significant attention. These results are remarkable not because they demonstrate unexpected design principles, but because these principles were learned directly from the data without domain knowledge.

8. DISCUSSION

From a modeling perspective, our experiments demonstrate the efficacy of our approaches for accurately estimating the transition probabilities in Figure 3(b). Punera and Meguru show that user interaction behavior can be highly personalized [30]. As a result, in removing personal information, we may have limited our modeling power. Nonetheless, the flexibility of our model means that, if available, this information can be easily encoded. Indeed, recent advances in feature hashing suggest that personalization of Farley-Ring models is feasible at scale [38]. At a smaller scale, we also could benefit from incorporating other information in the same search task. Just as user behavior may vary across individuals, behavior may vary depending on where user is in the search process.

Consider the case of a designer interested in using our models for developing a new SERP arrangement. The results of our first experiment suggest that, if we are only inter-

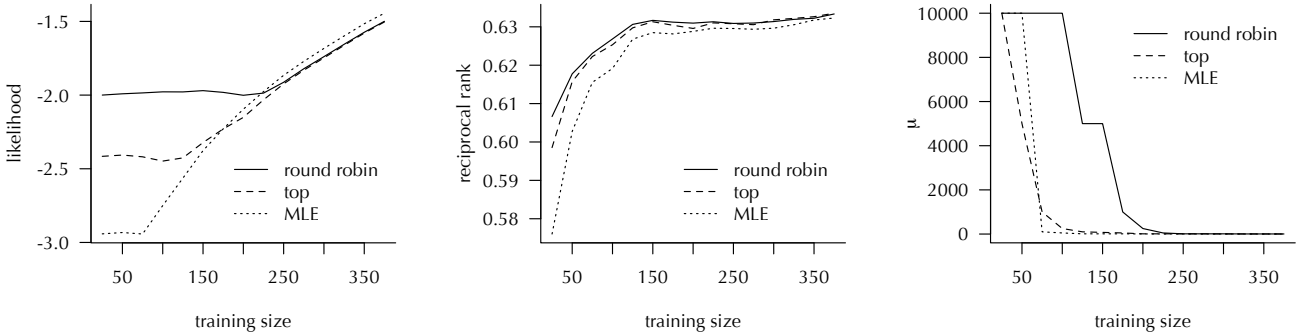
(a) Search Engine 1

	Experiment 1			Experiment 2		
	top	random	round robin	top	random	round robin
reciprocal rank	**0.5824**	0.5813	0.5671	0.4314	0.6093	**0.6116**
likelihood	**-2.1520**	-2.2224	-2.3702	-2.9429	-1.9877	**-1.9746**

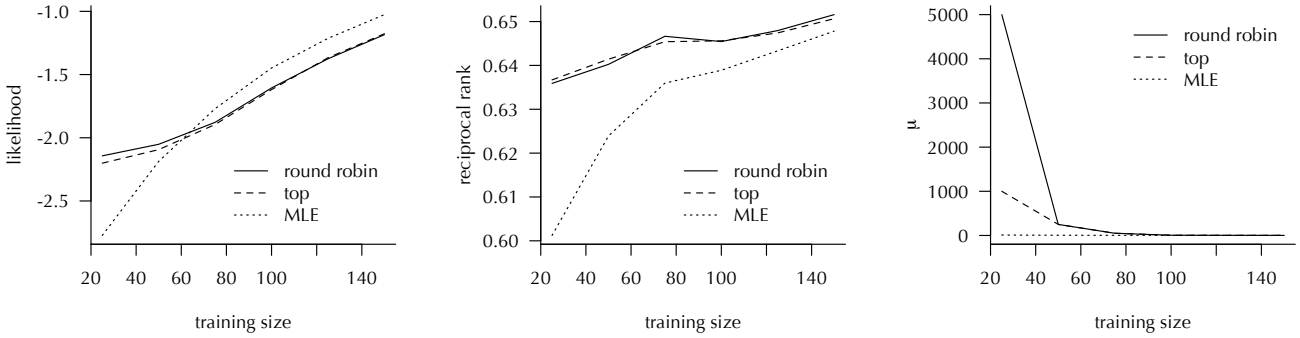
(b) Search Engine 2

	Experiment 1			Experiment 2		
	top	random	round robin	top	random	round robin
reciprocal rank	**0.6307**	0.6275	0.5980	0.6127	**0.6300**	0.6285
likelihood	**-1.9639**	-2.0702	-2.0212	-2.3292	-2.2672	**-2.0983**

Table 2: Experimental Results. Experiment 1: The evaluation set consists of sessions where the most frequent arrangement was presented to users. Experiment 2: The evaluation set consists of sessions where the least frequent arrangements were presented to users.

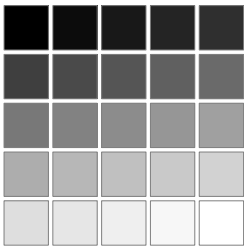


(a) Search Engine 1

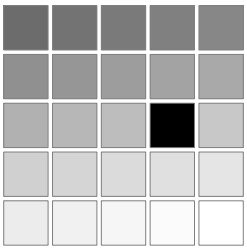


(b) Search Engine 2

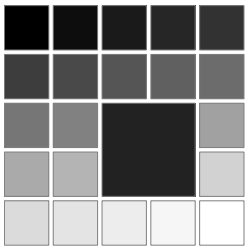
Figure 8: Experiment 3 Results: The evaluation set consists of sessions where arrangements of frequency 1000 were presented to users. The left and center graphs present performance with our metrics. The right graphs present the optimal value of the smoothing parameter μ on a separate validation set.



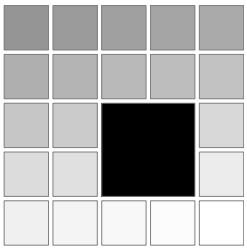
(a) Homogenous Modules



(b) One Image Module



(c) One Large Module



(d) One Large Image Module

Figure 9: Extrapolation experiments. Artificially-created arrangements were provided to our trained model. Visualizations reflect the probability of transitioning from the start state to the respective modules. Example layouts include: (a) 25 grid-oriented 'web result' modules, (b) 24 'web result' modules and one 'image' module, (c) 21 'web result' modules and one large 'web result' module, and (d) 21 'web result' modules and one large 'image' module. **Our model was trained using only standard web SERPs; no 5×5 layouts or modules of this shape were used in training the model.**

$\pi - \pi_{\text{planar}}$	module
0.0400	image vertical
0.0367	web 2
0.0362	web 3
0.0277	web 1
0.0202	web 10
0.0142	web 5
0.0083	web 6
0.0069	web 4
0.0044	web 7
-0.0004	web 9
-0.0014	web 8
-0.0049	query suggestion (inline)
-0.0142	query suggestion (left)
-0.0148	search history
-0.0220	pagination
-0.0533	result count
-0.0647	search box (bottom)

Table 3: Visualizing module importance by inspecting the stationary distribution of the Markov chain.

ested in modeling a particular arrangement, then the data collection and training on alternative arrangements might be unnecessary if we already have a great deal of data on the arrangement of interest. However, if there is no data, for example if the arrangement has not been exposed to any users, then the results of our second experiment suggest that the designer may benefit from a model based on round robin data. If there is a small amount of data, perhaps from some preliminary eye-tracking studies, then the results of our third experiment suggest that the designer may still benefit from a model based on round robin data. However, the value of this model will decay as we accumulate more data on the target arrangement.

Alternatively, consider the case of a large scale search engine analyst interested in using our models to retrospectively analyze user behavior on different automatic search engine arrangement decisions. The results of our first and third experiments suggest that, for very common arrangements, we only need to look at the raw data and use the maximum likelihood model. However, for most arrangements, we will have very little observed mousing data. Therefore, a model based on round robin sampling can be updated with the observed data in the logs. As a result, the analyst will have more reliable estimates of how users were behaving, even for tail arrangements. This is important to note since more than 99% of arrangements in our data set have fewer than 200 sessions (Figure 2).

We conclude our discussion by recalling our motivation to develop a model which could handle (a) inconsistent starting position, (b) nonmonotonic scan order, and (c) presentation bias. Our Markov model itself allows for each of these behaviors to be captured. The results of our base experiments demonstrate that we capture observed nonmonotonic transition probabilities. The attention deviation (Figure 3) and extrapolation (Figure 9) experiments confirm that we are addressing the inconsistent starting position and presentation bias.

9. CONCLUSION

We have studied the robustness of mouse-tracking models for web search. We paid particular attention to the development of models which could be applied to situations where little or no session data is available. We found that models based on round robin data performed best for when no target data was available. However, a hybrid model which updates a base Farley-Ring model with data from small scale studies performed best.

There are several directions of future research. First, we are interested in integrating our modeling framework with existing information retrieval evaluation metrics. The straightforward generalization of previous models should ease this process. Second, we believe that our task definition is novel and can benefit from further advancement of features (e.g. personalization, search task) and core algorithms (e.g. incorporating temporal information). Our models incorporate relatively simple geometric properties of modules. Much of the work in visual attention modeling can be incorporated as additional features to improve performance further. We are interested in comparing such a model to existing models which use only graphical features and eye-tracking data alone [6, 21]. Third, we are interested in studying the portability of our model to non-SERP pages such as portal pages and text-rich pages. We believe that further modeling insights can be found by conducting such studies.

Finally, we are interested more deeply evaluating the extrapolation ability of our models. This can be achieved by either conducting an eye-tracking evaluation or by assessing the usefulness of predictions as a tool for designers [33].

10. ACKNOWLEDGMENTS

This work would not have been possible without the support and feedback from Jaime Arguello, Peter Bailey, Georges Dupret, Luong Hoang, and Jeremy Hubert.

11. REFERENCES

[1] J. Arguello, F. Diaz, J. Callan, and J.-F. Crespo. Sources of evidence for vertical selection. In *SIGIR 2009*, 2009.

[2] E. Arroyo, T. Selker, and W. Wei. Usability tool for analysis of web designs using mouse tracks. In *CHI 2006*, 2006.

[3] R. Atterer, M. Wnuk, and A. Schmidt. Knowing the user's every move: user activity tracking for website usability evaluation and implicit interaction. In *WWW 2006*, 2006.

[4] M. Bennett and A. Quigley. Creating personalized digital human models of perception for visual analytics. In *UMAP 2011*, 2011.

[5] A. Broder, M. Ciaramita, M. Fontoura, E. Gabrilovich, V. Josifovski, D. Metzler, V. Murdock, and V. Plachouras. To swing or not to swing: learning when (not) to advertise. In *CIKM 2008*, 2008.

[6] G. Buscher, E. Cutrell, and M. R. Morris. What do you see when you're surfing?: using eye tracking to predict salient regions of web pages. In *CHI 2009*.

[7] O. Chapelle, D. Metzler, Y. Zhang, and P. Grinspan. Expected reciprocal rank for graded relevance. In *CIKM 2009*, 2009.

[8] O. Chapelle and Y. Zhang. A dynamic bayesian network click model for web search ranking. In *WWW 2009*, 2009.

[9] M. C. Chen, J. R. Anderson, and M. H. Sohn. What can a mouse cursor tell us more?: correlation of eye/mouse movements on web browsing. In *CHI 2001*.

[10] N. Craswell, O. Zoeter, M. Taylor, and B. Ramsey. An experimental comparison of click position-bias models. In *WSDM 2008*, pages 87–94, 2008.

[11] G. Dupret and C. Liao. A model to estimate intrinsic document relevance from the clickthrough logs of a web search engine. In *WSDM 2010*, 2010.

[12] J. U. Farley and L. W. Ring. A stochastic model of supermarket traffic flow. *Operations Research*, 14(4):555–567, July/August 1966.

[13] Q. Guo and E. Agichtein. Exploring mouse movements for inferring query intent. In *SIGIR 2008*, 2008.

[14] Q. Guo and E. Agichtein. Ready to buy or just browsing?: detecting web searcher goals from interaction data. In *SIGIR 2010*, 2010.

[15] Q. Guo and E. Agichtein. Towards predicting web searcher gaze position from mouse movements. In *CHI 2010*, 2010.

[16] Y. He and K. Wang. Inferring search behaviors using partially observable markov model with duration (pomd). In *WSDM 2011*, 2011.

[17] J. Huang, R. W. White, and G. Buscher. User see, user point: Gaze and cursor alignment in web search. In *CHI 2012*, 2012.

[18] J. Huang, R. W. White, G. Buscher, and K. Wang. Improving searcher models using mouse cursor activity. In *SIGIR 2012*, 2012.

[19] J. Huang, R. W. White, and S. Dumais. No clicks, no problem: using cursor movements to understand and improve search. In *CHI 2011*, 2011.

[20] T. Joachims. Optimizing search engines using clickthrough data. In *KDD 2002*, 2002.

[21] T. Judd, K. A. Ehinger, F. Durand, and A. Torralba. Learning to predict where humans look. In *ICCV*, pages 2106–2113, 2009.

[22] J. Li, S. Huffman, and A. Tokuda. Good abandonment in mobile and pc internet search. In *SIGIR 2009*, 2009.

[23] T.-Y. Liu. *Learning to Rank for Information Retrieval*. Springer, 2011.

[24] L. McCay-Peet, M. Lalmas, and V. Navalpakkam. On saliency, affect and focused attention. In *CHI 2012*.

[25] V. Navalpakkam and E. Churchill. Mouse tracking: Measuring and predicting users' experience of web-based content. In *CHI 2012*, 2012.

[26] V. Navalpakkam, L. Jentzsch, R. Sayres, S. Ravi, A. Ahmed, and A. Smola. Measurement and modeling of eye-mouse behavior. In *WWW 2013*, 2013.

[27] V. Navalpakkam, J. Rao, and M. Slaney. Using gaze patterns to study and predict reading struggles due to distraction. In *CHI 2011*, 2011.

[28] U. Ozertem, O. Chapelle, P. Donmez, and E. Velipasaoglu. Learning to suggest: a machine learning framework for ranking query suggestions. In *SIGIR 2012*, 2012.

[29] D. Parkhurst, K. Law, and E. Niebur. Modeling the role of salience in the allocation of overt visual attention. *Vision Research*, 42(1):107–23, January 2002.

[30] K. Punera and S. Merugu. The anatomy of a click: modeling user behavior on web information systems. In *CIKM 2010*, 2010.

[31] S. Robertson. The probability ranking principle. *Journal of Documentation*, 1977.

[32] K. Rodden, X. Fu, A. Aula, and I. Spiro. Eye-mouse coordination patterns on web search results pages. In *CHI 2008*, 2008.

[33] R. Rosenholtz, A. Dorai, and R. Freeman. Do predictions of visual perception aid design? *ACM Trans. Appl. Percept.*, 8:12:1–12:20, February 2011.

[34] R. Rosenholtz, Y. Li, J. Mansfield, and Z. Jin. Feature congestion: a measure of display clutter. In *CHI 2005*.

[35] C. J. van Rijsbergen. *Information Retrieval*. Butterworths, 1979.

[36] C. Wang, Y. Liu, M. Zhang, S. Ma, M. Zheng, J. Qian, and K. Zhang. Incorporating vertical results into search click models. In *SIGIR 2013*, 2013.

[37] K. Wang, N. Gloy, and X. Li. Inferring search behaviors using partially observable markov (pom) model. In *WSDM 2010*, 2010.

[38] K. Weinberger, A. Dasgupta, J. Langford, A. Smola, and J. Attenberg. Feature hashing for large scale multitask learning. In *ICML 2009*, 2009.