Estimating Rates of Rare Events at Multiple Resolutions

Deepak Agarwal,Andrei Broder,Deepayan Chakrabarti, Dejan Diklic,Vanja Josifovski,Mayssam Sayyadian
Yahoo! Research
Sunnyvale,CA,USA
{dagarwal, broder, deepay, dejand, vanjaj, mayssam}@yahoo-inc.com

ABSTRACT

We consider the problem of estimating occurrence rates of rare events for extremely sparse data, using pre-existing hierarchies to perform inference at multiple resolutions. In particular, we focus on the problem of estimating click rates for (webpage, advertisement) pairs (called *impressions*) where both the pages and the ads are classified into hierarchies that capture broad contextual information at different levels of granularity. Typically the click rates are low and the coverage of the hierarchies is sparse. To overcome these difficulties we devise a sampling method whereby we analyze a specially chosen sample of pages in the training set, and then estimate click rates using a two-stage model. The first stage imputes the number of (webpage, ad) pairs at all resolutions of the hierarchy to adjust for the sampling bias. The second stage estimates click rates at all resolutions after incorporating correlations among sibling nodes through a tree-structured Markov model. Both models are scalable and suited to large scale data mining applications. On a real-world dataset consisting of 1/2 billion impressions, we demonstrate that even with 95% negative (non-clicked) events in the training set, our method can effectively discriminate extremely rare events in terms of their click propensity.

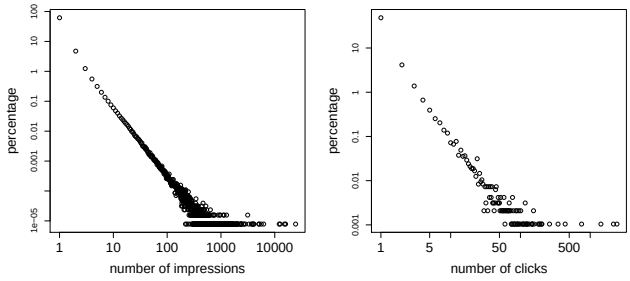
Categories and Subject Descriptors: H.1[Information Systems]: Models and Principles

General Terms: Algorithms

Keywords: Imputation, Hierarchy, Clickthrough Rate, Tree-structured Markov model, Maximum Entropy, Internet Advertising, Algorithmic Advertising

1. INTRODUCTION

Web advertising supports a large swath of today's Internet ecosystem with an estimated $15.7 billion in revenues for 2005 (`www.cnnmoney.com`). Some of these ads are textual and some are graphical. *Contextual advertising* or *Content Match* (CM) refers to the placement of commercial textual advertisements within the content of a generic web page, while *Sponsored Search* (SS) advertising consists in placing ads on result pages from a web search engine, with ads driven by the originating query. In contextual advertising usually there is a commercial intermediary, called an *ad-network*, in charge of optimizing the ad selection with the twin goal of increasing revenue (shared between publisher and ad-network) and improving user experience. Typically the ad-network and the publisher are paid only when the user *clicks* on an advertisement.



(a) Impression (b) Clicks

Figure 1: ***(a) Distribution of impression events and (b) click events. Plots are on log-log scale but ticks are on the original scale. 99.7% of impression events had no clicks.***

In this paper we examine data generated by a content match system where every showing of an ad on a webpage (called an *impression*) constitutes an event. Some pages are much more popular than others and generate many more impressions, but for all impressions the click rate is at most a few percent. Figure 1(a) shows the frequency of (page, ad) pairs and Figure 1(b) shows the same distribution for a subset of impressions where a user clicks on the ad being shown on the page. Clearly, an overwhelming majority of (page, ad) pairs are extremely rare while a small fraction account for a large fraction of total impressions and clicks.

Finding ads suitable to a given page depends on many factors including: the page content, the publisher and ad-network business aims, the target audience for the advertisers, and the experience of the web user [14]. Estimating clicks per impression, or click-through rates (CTRs, henceforth), for a (page, ad) pair is a very important tool in finding such relevant ads for webpages. However, rate estimation is difficult due to sparsity in the impression distribution and rarity of clicks. Sparseness is pervasive since a large fraction of webpages and ads tend to be ephemeral. This occurs because webpages are often modified, content is generated dynamically, and ads are updated on a regular basis. Naive statistical estimators based on frequencies of event occurrences incur high statistical variance and fail to provide satisfactory predictions, especially for rare events. The usual procedure involves either removing or aggregating rare events to focus on the frequent ones [11]. While this might help in estimation at the "head" of the curve, the loss

KDD '07 San Jose, CA USA

in information leads to poor performance at the "tail." In Internet advertising, the tail accounts for several billion dollars annually, making reliable CTR estimation an important problem.

We describe a statistical method to estimate rates of rare events in scenarios where these events are organized in an existing, well understood, and intuitive hierarchy. Examples abound in the online-advertising domain and beyond: users organized in a geographic hierarchy, movies organized by genres, books organized by subject matter, webpages and ads organized by semantic content, disease symptoms organized by syndromic groups, and so on. The hierarchical structure is expected to induce correlations among sibling nodes at every level of the hierarchy. Can we exploit such correlations to improve our estimates? We demonstrate that the answer is in the affirmative in our application of Internet advertising.

Classical machine learning approach converts each page and ad into a set of features and learns a model to predict CTRs. However, the number of features required in content match is large (we consider page features like URL, title, HTML tags, words, etc.; ad features like title, bidded phrases, the "landing" webpage of the advertiser, and so on). Aggregating results from such models to estimate CTRs at coarser resolutions generally incurs high variance. We note that the simple strategy of using hierarchies as features fails to incorporate correlations that exist among siblings nodes.

In this paper, we propose a method to estimate rates of rare events at multiple resolutions, as determined by existing hierarchies for both pages and ads. These have been created and constantly refined by domain experts; they are well understood and routinely used in applications. Apart from ease of interpretability, there are other advantages in using such hierarchies. First, they provide a natural and intuitive framework to analyze CTRs at multiple scales, from coarser to successively finer resolutions. The hierarchies induce a tree structure on the CTRs, that is, CTRs of page and ad node pairs from finer resolutions are nested within node-pairs from coarser resolutions. The structure is expected to induce correlations in CTRs among siblings at each resolution. Also, at coarser resolutions, aggregation helps in combating data sparseness and providing *reliable* estimates for rare events. Note that aggregation can substantially reduce variance but is likely to incur bias. However, if the hierarchy is informative and siblings are homogeneous, the variance reduction will far outweigh the increase in bias.

Such multi-resolution estimates are extremely useful and serve several purposes. First, the multi-resolution modeling framework provides better estimates at the finest resolution by "borrowing" information from coarser resolutions. In fact, coarser resolution estimates act as priors that influence estimates at finer resolutions; the amount of influence depending on data sparsity. Second, the ability to estimate CTRs with precision at multiple scales helps in discovering macro level patterns that are hard to estimate reliably from feature based models, especially with sparse data. Among other things, the macro level information can be used to track the system through time for emerging trends, sudden changes, etc. In fact, such estimates can also help the user design efficient and targeted online experiments to explore opportunities that could be potentially lucrative (see [13] for an example.)

We assume that there exist classifiers that can accurately classify a page (ad) to a unique path in the page (ad) hierarchy. However, classification of a page into the hierarchy require features that are obtained by crawling the page. Crawling is an expensive operation that induces cost on both the ad-network and the page publishers in terms of network bandwidth, storage and computational resources. Furthermore, as we operate retrospectively on historical data, some URLs in the logs do not exist any more and we cannot crawl them, e.g., dynamic webpages, or webpages requiring user authorization. We restrict our study to pages that can be crawled. Further, to reduce computation time and load on the system, we crawl only a sample of the webpages that receive impressions, and project our results over the entire set of events. In fact, sampling from majority class is a standard practice when predicting a rare response[9]. Moreover, we show that the results are not significantly affected over a wide range of sampling fractions.

However, sampling introduces bias that needs to be adjusted for in the estimation process. In traditional classification or regression problems with rare response, this is done by an adjustment to the weight of observations. Such strategies don't work in our case since we are estimating a response at multiple resolutions in a hierarchy. We propose a two stage model where the first stage model estimates impressions at all resolutions after correcting for the sampling bias. Conditional on the estimated impressions, our second stage model estimates the CTRs at all levels of the hierarchy through a tree-structured Markov model. Both models exploit the hierarchical structure and are scalable.

1.1 Our contributions

We provide a method to estimate rates of rare events at multiple resolutions, where these resolutions are derived from a pre-existing hierarchy. We illustrate our method on a large-scale Content Match application. A sampling-based approach is used to reduce crawling costs. We then present a two-stage modeling approach to estimate CTRs at all resolutions. The first stage algorithm imputes impression volume at all resolutions after adjusting for the sampling bias. The imputation is implemented through a simple iterative proportional fitting algorithm (IPF). Conditional on the imputed impressions, a tree-structured Markov model estimates the click rates after accounting for correlations induced by the hierarchical structure. Our methods are highly scalable, and we empirically demonstrate the efficacy of our approach in estimating rare click rates at finer resolutions by using reliable estimates at coarser resolutions. Through validation experiments, we show that our approach can accurately predict rates of events that do not even occur in the training data by utilizing structure present in the hierarchies.

The paper is organized as follows. An overview of our method is given in Section 2, and details in Section 3. Detailed experimental results are shown in Section 4. We discuss related work in Section 5 and finally conclude in Section 6.

2. OVERVIEW

We analyze clicks and impressions from a subset of historical logs of a current Content Match system. The logs contain a large number of (page, ad) impressions, a small fraction of which generate clicks. Pages and ads are classified into a pre-existing hierarchy where the nodes correspond to broad contextual themes (e.g., skiing $\subset$ winter sports $\subset$ sports). We estimate CTRs using a two stage procedure. First, we crawl a sample of URLs from the logs and impute impression volume at all resolutions. Then, we estimate CTRs at these resolutions, using clicks and imputed impression volumes. Before describing our method, we first lay out our notation.

Let i and j represent nodes from the page and ad hierarchies respectively, and let ij denote an arbitrary element the cross-product of the two hierarchies. We shall refer to the elements in the cross-product as *regions*. In general, regions may overlap with each other, but in this paper, we only consider estimation for a set of regions $\mathcal{Z}$ that form a tree. In other words, for any two regions $r_1, r_2 \in \mathcal{Z}$, one and only one of the following is true: $r_1 \subset r_2$, $r_2 \subset r_1$, $r_1 = r_2$ or $r_1 \cap r_2 = \phi$. For ease of exposition, we further simplify our problem and assume that the page and ad hierarchies are identical (as is indeed the case in our illustrative application), and

	Clicked pool	*Sampled non-clicked pool*
Crawlable pool	$P_{c,c}$	$P_{s,c}$
Uncrawlable pool	$P_c - P_{c,c}$	$P_s - P_{s,c}$
Total	P_c	P_s

Table 1: *Page distribution after sampling.*

only consider regions ij where both i and j belong to the same depth in the hierarchy. Let the hierarchy be a tree with $L+1$ levels, with the root at depth 0 and leaves at depth $L > 0$. For any region $r \in \mathcal{Z}$, let $d(r)$ represent its depth in the tree, and $pa(r)$ its parent region in the tree. Denote by $\mathcal{Z}^{(\ell)} \in 2^{\mathcal{Z}}$ the set of all regions obtained by considering combinations of nodes in the page and ad hierarchies at depth $\ell(\ell = 0, \cdots, L)$. Note that $\mathcal{Z}^{(0)}$ is the root and consists of one region, and all other regions are subsets of $\mathcal{Z}^{(0)}$. The importance of the hierarchy can be stated as follows: both the impression volumes and CTRs for regions in $\mathcal{Z}^{(\ell+1)}$ with the same parent in $\mathcal{Z}^{(\ell)}$ are expected to be correlated. Accounting for such correlations would induce smoothness in the estimation process and is expected to reduce the overall mean squared error. This is similar in spirit to time series and spatial modeling where accounting for autocorrelations that are induced due to proximity in time or space have an impact on overall predictive performance [2, 8].

2.1 Sampling and imputation

Obtaining features for both ads and pages is necessary for classification into their respective hierarchies. Features on ads are readily available from historical records, so all ads can be classified and the exact impression volume for every node in the ad hierarchy can be computed. However, features on pages require crawling. As discussed in Section 1, it is not possible to crawl all pages and hence we restrict ourselves to those that could be crawled. Also, since crawling is an expensive operation, we further reduce cost by crawling only a sample of pages from this restricted set. Our goal is to estimate the impression volume for every region $r \in \mathcal{Z}$ after adjusting for this sampling bias.

Sampling.

We define a page to be clicked if it has received at least one click in our data, and to be crawlable if crawling the page does not lead to an error (e.g., "Page not found" or "Authorization required"). Since the number of pages P_c in the clicked pool is relatively small, we attempt to crawl all of them, resulting in $P_{c,c}$ crawlable pages. From the majority class of non-clicked pages, we take a random sample of size P_s, of which $P_{s,c}$ are crawlable. In fact, $\kappa = (P_{c,c} + P_{s,c})/(P_c + P_s)$ provides an estimate of the fraction of crawlable pages (see table 1). Scaling all impressions volumes in the ad hierarchy (which are known) by κ provides corresponding estimates conditioned on the crawlable pool. Henceforth, unless otherwise mentioned, all our inferences are conditional on the crawlable pages. For each crawled page, we include all (page, ad) impressions associated with that page from historical data, and map them to the corresponding regions in $\mathcal{Z}$. This yields the number of sampled impressions in each region $r \in \mathcal{Z}$.

Imputation of impression volumes.

Given the sampled impressions, our goal is to estimate the true number of impressions for all regions of interest in $\mathcal{Z}^{(\ell)}(\ell = 0, \cdots, L)$. Since all ad classifications are known, we know the total number of impressions at each node in the ad hierarchy. To obtain an unbiased estimate of the total number of impressions for the page hierarchy nodes, the sampled impression totals at each level of the page hierarchy are scaled up by a constant factor, this factor being selected so that the total number of impressions match the total impressions in the historical data at each level. We also obtain a lower bound on impression volume for each region, which is the total number of impressions in the region obtained from our sample. Thus, we have region totals that are prone to sampling variability and marginal totals that are based on a much larger set of observations and have smaller variance. How do we distribute the excess impressions missed by the sampling process among the regions? First, by forcing the estimates to conform as closely as possible to the more accurate marginal impression totals at each node in the page and ad hierarchies, we reduce the overall variance of our region estimates.[10] considered a similar problem in a different context for 2×2 tables. Second, by definition, the sum of estimated impressions for children regions nested within a parent region should agree with that of the parent. We accomplish these by *imputing* excess impressions using a maximum entropy formulation, subject to the constraints mentioned above. Section 3 provide the details.

2.2 Estimating click-through rates (CTRs)

The second stage model is used to obtain CTR estimates in all regions, conditional on the imputed impression volumes. Denoting the exact number of clicks in a region r by c_r and the imputed number of impressions by $\hat{N}_r$, the maximum likelihood estimate (MLE) for the CTR λ_r is given by $\hat{\lambda}_r = c_r/\hat{N}_r$. The variability in this estimator gets higher with smaller sample size. Thus, a ratio of $1/10$ is less reliable than $10/100$. Our method smoothes the MLEs using the hierarchical structure of the regions: regions sharing the same parent are expected to share some common characteristics and hence similar λ_r values. We exploit this correlation by modeling the CTRs through a multi-resolution tree-structured Markov model [5, 7].

The central idea of the model is as follows. We assign a state variable to each node in $\mathcal{Z}$. Conditional on the states, the observed CTRs are independent. Smoothing of CTR estimates is accomplished through a Markovian model on the states. In particular, we assume that the states of children sharing the same parent are drawn from a distribution centered around the state of the parent. These sequences of recursive one-step Markovian distributions defined in a bottom-up fashion specify a joint distribution on the entire state space of CTR values. The posterior distribution of the state variables given the data provides the smoothed CTR estimates. Note that although the observed CTRs are conditionally independent, unconditionally they are not: The Markovian structure on the states induce dependencies in the observations.

An attractive feature of the model is its efficient computational aspect. For known values of all variance components, the posterior of the states can be computed by a Kalman filter algorithm that performs a *filtering* step in a bottom-up fashion from the leaves to the root followed by a *smoothing* step in a top-down fashion from the root to the leaves. However, since the variance components associated with the model are unknown, we estimate them through an Expectation-Maximization (EM) algorithm [1]. The EM algorithm involves iterating through the filtering and smoothing steps several times (for our application, less than 25) until convergence.

3. MODELING DETAILS

We now describe the first and second stage models in detail. The first stage model corrects for the sampling bias in imputing impression volumes, followed by the second-stage model that estimates CTRs at multiple resolutions.

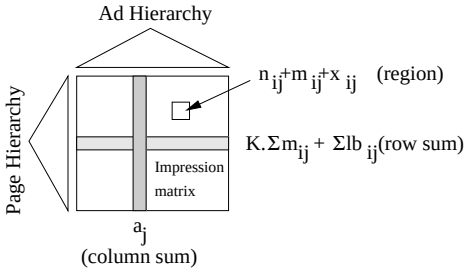


Figure 2: *Impressions in a region, row, and column for $\mathcal{Z}^{(2)}$.*

3.1 First Stage: Imputation

For the sake of notational simplicity, we assume that the set of regions $\mathcal{Z}$ consists of two successive levels of nested regions corresponding to depths 1 and 2 respectively. Generalization to all regions formed by the full page and ad hierarchies follows easily. Let IJ and ij denote regions in $\mathcal{Z}^{(1)}$ and $\mathcal{Z}^{(2)}$ respectively. Let n_r and m_r denote impressions in region r from the clicked and sampled non-clicked pools of webpages. Thus, $lb_r = n_r + m_r$ provides a lower bound on the impression volume for region r. Let N_r denote the true impression volume in region r that is to be estimated. Consider the linear transformation $x_r = N_r - lb_r$. We solve our estimation problem in terms of x_rs and derive estimates of N_r as $\hat{N}_r = \hat{x}_r + lb_r$, where $\hat{x}_r$ is our estimate of x_r. In fact, one can interpret x_r's as *excess* impressions to be allocated to adjust for the sampling bias. We split the subsequent discussion into three parts: (a) data preparation and consistency, (b) constraints on the imputation model, and (c) using the constraints to estimate the impression volumes.

Data preparation and consistency.

A page (ad) classified to a node i in the taxonomy is assumed to belong to the entire path from i to the root. Also, a page (ad) may get classified to a node at depth other than L (i.e., leaf level). However, this may create inconsistencies in the total number of impressions and clicks obtained at different levels. For instance, the total number of impressions (clicks) for a group of children regions may be strictly smaller than the impressions (clicks) of the parent region they are nested within. To ensure consistency, we uniformly distribute the extra impressions and clicks in a parent node among its children. The steps are repeated at every level in a top-down fashion. Thus, each impression in a non-leaf region is guaranteed to come from some smaller region nested within it.

Constraints on the imputation model.

The region impressions are estimated subject to certain *linear* constraints. We impose three sets of constraints, all of which are expressed in terms of the excess impressions x_r (Figure 2). The first set of constraints, called *column constraints*, ensures that the sum of impressions along a column adds up to the total impressions for the corresponding node in the ad hierarchy:

$$\sum_i x_{ij} = a_j - \sum_i lb_{ij} = CS_j^{(2)}; \quad \text{for all } j \text{ in Level 2} \qquad (1)$$

$$\sum_I x_{IJ} = a_J - \sum_I lb_{IJ} = CS_J^{(1)}; \quad \text{for all } J \text{ in Level 1} \qquad (2)$$

where $a_j(a_J)$ is the total impression volume for node $j(J)$ in the ad hierarchy, and $CS_\cdot^{(\cdot)}$ represents the excess impressions in the column that were missed by the sampling process. Note that for a node J at level 1 in the ad taxonomy, $a_J = \sum_{j:pa(j)=J} a_j$, where $pa(j)$ denotes the parent of node j, that is, the column impressions total for a level 1 node is the sum of the column totals of its children in level 2. Also, $\sum_j CS_j^{(2)} = \sum_J CS_J^{(1)} = TotExcess$, where $TotExcess$ is the total number of excess impressions in the data. The second set of constraints, called *row constraints*, preserve impression volumes at nodes in the page hierarchy and are given as follows:

$$\begin{aligned} \sum_j x_{ij} &= K^{(2)} \sum_j m_{ij} = RS_i^{(2)}; \qquad \forall i \\ \sum_J x_{IJ} &= K^{(1)} \sum_J m_{IJ} = RS_I^{(1)}; \qquad \forall I \end{aligned} \qquad (3)$$

where $RS_\cdot^{(\cdot)}$ represents the excess impressions aggregated for each node in the page taxonomy, and $K^{(1)}$ and $K^{(2)}$ are constants for levels 1 and 2. The underlying assumption is that for each sampled impression, there are $K^{(\cdot)}$ times as many excess impressions from the non-clicked pool that did not appear in the sample. Since we randomly sample pages from the non-clicked pool, this simple adjustment is reasonable. The constants $K^{(\cdot)}$ are chosen to preserve total impression volume, i.e., we choose them so that $\sum_i RS_i^{(2)} = \sum_I RS_I^{(1)} = TotExcess$. Our third set of constraints, called *block constraints*, ensure that the excess impressions allocated to a region at level 1 equals the sum of excess impressions allocated to regions nested within it at level 2:

$$\sum_{ij:pa(ij)=IJ} x_{ij} = x_{IJ}; \quad \text{for all } IJ \qquad (4)$$

We note that the true impression volumes satisfy the block constraints, hence it is necessary to impose them in our imputation. Analogous row, column and block constraints are imposed at all other levels $\ell(\ell = 0, \cdots, L)$.

Estimating impression volumes.

Given a set of positive initial prior values $\{x_r(0)\}$ for all regions $r \in \mathcal{Z}$, we want a solution $\{x_r\}$ which is as close as possible to the prior $\{x_r(0)\}$ but satisfies all the row, column and block constraints. This is equivalent to finding a solution that has the smallest discrepancy from the prior in terms of Kullback-Leibler divergence, subject to the linear constraints [6]. It is also referred to as the Maximum Entropy model, since when $\{x_r(0)\}$ is uniform, the solution maximizes the Shannon entropy.

We solve this Maximum Entropy imputation problem using an Iterative Proportional Fitting (IPF) algorithm [6], which iterates cyclically over all the constraints and updates the x_r values to match the constraints as closely as possible. Specifically, at the t^{th} iteration, suppose a constraint of the form $\sum_r k_r x_r = C$ is being violated ($k_r = 0$ or 1 for all our constraints). Let the current value $C(t)$ of the LHS be $C(t) = \sum_r k_r x_r(t)$, where $C_t \neq C$. Then, IPF adjusts each element x_r involved in the constraint by a constant factor $C/C(t)$ to get the new values $x_r(t+1) = x_r(t) \cdot C/C(t)$. Note that this update rule ensures non-negativity of the final solution. Such updates are performed for all constraints until convergence. Our algorithm jointly estimates all x_rs by iterating through a series of top-down and bottom-up scalings. Here, we provide a description for a two level tree; a complete description of the algorithm for an arbitrary number of levels is provided in Figure 3. At the t^{th} iteration, we start with level 1, and modify $\{x_{IJ}(t)\}$ to $\{x_{IJ}(t+1)\}$ after adjusting for the row and column constraints. This changes the values of $\{x_{ij}(t)\}$s at level 2 to $\{x_{ij}^*(t)\}$s by adjusting for the corresponding block constraints. Now, we switch

Initialization:

Begin with a prior $\{x_r(0)\}$ for regions $r \in \mathcal{Z}^{(1)}$ of level 1

From iteration t to $t+2$:

Begin Top-down:

For all $r \in \mathcal{Z}^{(1)}$, $x_r(t) \rightarrow$ row constraints $\rightarrow$ column constraints $\rightarrow x_r(t+1)$

For levels $\ell = 2, \cdots, L$

For all $r \in \mathcal{Z}^{(\ell)}$: $x_r(t) \rightarrow$ block constraints with $x_{pa(r)}(t+1)$ on the RHS $\rightarrow x_r^*(t)$, where $pa(r)$ is the parent region subsuming r

$x_r^*(t) \rightarrow$ row constraints $\rightarrow$ column constraints $\rightarrow x_r(t+1)$

Begin bottom-up:

For all $r \in \mathcal{Z}^{(L)}$, $x_r(t+2) = x_r(t+1)$

For levels $\ell = L-1, \cdots, 1$

For all $r \in \mathcal{Z}^{(\ell)}$: $x_r^*(t+1) = \sum_{k \in ch(r)} x_k(t+1)$, where $ch(r)$ are all children regions nested within r

$x_r^*(t+1) \rightarrow$ row constraints $\rightarrow$ column constraints $\rightarrow x_r(t+2)$

Iterate until convergence, i.e., until all constraints are satisfied upto a user-defined accuracy factor

Figure 3: The Iterative Proportional Fitting (IPF) algorithm for imputing impressions

to level 2, and change the $\{x_{ij}^*(t)\}$s to $\{x_{ij}(t+1)\}$s by adjusting for row and column constraints. This completes the top-down step. In the bottom-up step, the leaf regions (in this example, regions at depth 2) do not change, i.e., $x_{ij}(t+2) = x_{ij}(t+1)$. Using the block constraints, the values at level 1 change to $x_{IJ}^*(t+1) = \sum_{ij:pa(ij)=IJ} x_{ij}(t+2)$ followed by row and column scalings to satisfy the level-1 constraints, ending with $x_{IJ}(t+2)$. The top-down and bottom-up steps are iterated until convergence. In all our experiments, the algorithm converges rapidly, requiring at most 156 iterations for an error tolerance of 1%.

One variable in our imputation algorithm is the choice of prior. We assume $x_r(0) \propto lb_r$; this ensures that we distribute the excess impressions in proportion to the lower bounds obtained from the crawled sample as closely as possible subject to the linear constraints. An alternative is to simply use the traditional IPF algorithm, which starts with a prior of $x_r(0) \propto 1$, and computes the x_r values for each level separately, using only the row and column constraints. It can be shown that this automatically satisfies the block constraints as well, due to the relationships between the row and column sums at different levels. However, the prior distributes the excess impressions using an independence model and does not incorporate the a-priori interaction information we have in the lower bounds. We show empirically in Section 4 that the lower-bound prior outperforms the naive scheme significantly, for regions at all levels.

3.2 Second Stage: Rare Rate Modeling

This section is organized as follows. We first discuss data transformation used to facilitate our analysis. This is followed by a description of the tree-structured Markov model used to obtain smoothed estimates of CTRs. Finally, model-fitting via an EM algorithm is described.

Data transformation.

The distribution of raw CTRs is extremely skewed and the variance depends on the mean (roughly, $Var \propto mean/\hat{N}_r$), as shown later in Figure 6. Instead of modeling our data on the original scale, we use a transformed scale. A squared-root transform is commonly recommended for count data but we use the more stable Freeman-Tukey transformation [8], defined as follows:

$$y_r = \frac{1}{2}\left(\sqrt{\frac{c_r}{\hat{N}_r}} + \sqrt{\frac{c_r+1}{\hat{N}_r}}\right), \tag{5}$$

where c_r is the number of clicks in region r, and $\hat{N}_r$ is the imputed number of impressions. This transformation has a number of advantages, especially when modeling rare rates [8]. The second term provides a way to distinguish between zeros on the basis of the number of impressions, e.g., zero clicks from 100 impressions corresponds to a smaller transformed CTR than zero clicks from only 10 impressions. It also tends to symmetrize the otherwise extremely skewed rate distribution. Most important is its variance stabilization property, which makes the variance of the distribution independent of the mean (roughly, $Var \propto 1/\hat{N}_r$). This holds for our data as shown later in Figure 7.

Generative Model.

We describe the tree-structured Markov model used to model our data. In our context, the set of regions $\mathcal{Z}$ forms a tree structure, and we model the data through a generative model wherein the CTRs of the regions are connected to each other at different resolutions in a Markovian fashion. Each region r has a transformed rate y_r which is observed, a covariate vector u_r, and a latent state S_r. The covariate vector can represent any region-specific information, such as prior knowledge distinguishing different region "types." Examples might include percentage of total visits distributed by geography.

For simplicity, we assume a single covariate per level. In fact, in our example dataset, $u_r^T = 1$ for all r giving us one covariate for each level in the region hierarchy. Conditional on the states $\{S_r\}$ being known, we assume the observations y_r to be independently distributed as a Gaussian:

$$y_r | S_r, \beta^{(d(r))} \sim N(u_r^T \beta^{(d(r))} + S_r, V_r), \tag{6}$$

where $\beta^{(d(r))}$ is the unknown coefficient vector attached to covariates at level $d(r)$, and V_r is the unknown variance parameter. Intuitively, the latent S_r variables are adjusting for effects that are not accounted for by the covariates. However, estimating one S_r per region leads to severe overfitting; hence smoothing on S_rs is necessary. We perform smoothing by exploiting dependencies induced by the tree structure of regions:

$$S_r = S_{pa(r)} + w_r, \tag{7}$$

where $w_r \sim N(0, W_r)$ for all $r \in \mathcal{Z} \setminus \mathcal{Z}^{(0)}$. Also, w_r is independent of $S_{pa(r)}$ and $S_{Root} = W_{Root} = 0$. Figure 4 shows the model in graphical notation.

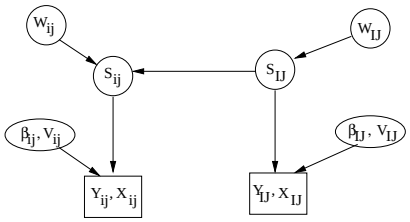


Figure 4: ***Generative model for two levels***

Since it is not feasible to estimate a separate W_r and V_r for each region, we need to make some assumptions on their structure. In our case, we assume that all regions at the same level have the same W_r value: $W_r = W^{(\ell)}$ for all $r \in \mathcal{Z}^{(\ell)}$. Modeling assumptions on V_r depend on the data and the tree structure of regions. In our case, $Var(y_r) \propto 1/\hat{N}_r$ (from Equation 5 and [8]). Hence, we assume that there is a V such that $V_r = V/\hat{N}_r$ for all $r \in \mathcal{Z}^{(\ell)}$. The ratios W_r/V_r determine the amount of smoothing that takes place with our model. Intuitively, if W_r is large relative to V_r, the sibling S_rs are drawn from a distribution that has high variance and hence we achieve little smoothing. In fact, if $W_r/V_r \to \infty$, then $S_r \to (y_r - u_r^T \beta^{(d(r))})$ and we perfectly fit the training data. On the other extreme, if $W_r/V_r \to 0$, then $S_r \to 0$ and we fit a regression model given by the covariates, with the maximum possible smoothing. As we show in Section 4, the smoothing achieved by this method performs well on our real-world dataset.

Correlations implied by the model.

To understand the dependency structure imposed by the model, we examine the correlations implied by the state equations. From Equation 7 and the independence of w_r and $S_{pa(r)}$, it follows that

$$Var(S_r) = \sum_{i=1}^{d(r)} W^{(i)}. \tag{8}$$

Thus, the variance in the states S_r depends only on the depth of region r, and increases as we move from coarser to finer resolutions. Also, for any two regions $r1$ and $r2$ at depth ℓ sharing a common ancestor q at depth $\ell' < \ell$, the covariance between the state values is given by $Cov(S_{r1}, S_{r2}) = Var(S_q)$, which depends only on ℓ'. Hence the correlation coefficient of nodes at level ℓ whose least common ancestor is at level ℓ' is given by

$$Corr(\ell, \ell') = \frac{\sum_{i=1}^{\ell'} W^{(i)}}{\sum_{i=1}^{\ell} W^{(i)}} \tag{9}$$

which depends only on the level of the regions and the distance to their least common ancestor. We note that y_rs are independent conditional on S_rs; however the dependencies in S_rs impose dependencies in the marginal distribution of y_rs.

Model-fitting by EM and Kalman filtering.

Model fitting is accomplished through an EM algorithm that estimates the posterior distribution of $\{S_r\}$s and $\{\beta^{(d(r))}\}$s, and also provides point estimates of the variance components $\{W^{(\ell)}\}$ and V. Next, we provide the core ideas of the algorithm, with details presented in the appendix.

The heart of the algorithm is a Kalman filtering step which efficiently estimates the posterior distribution of $\{S_r\}$s for fixed values of the variance components. The Kalman filtering algorithm itself consists of two steps, namely, a *filtering* step that aggregates information from the leaves up to the root, followed by a *smoothing* step that propagates the aggregated information in the root downwards to the leaves. The former collects information from the children and passes it to the parents while the latter passes information from parents to children. To provide intuition on the filtering step, note that we can invert the state equations to express parent states in terms of their children states:

$$\begin{aligned} S_{pa(r)} = &\quad E(S_{pa(r)}|S_r) &+(S_{pa(r)} - E(S_{pa(r)}|S_r)) \\ = &\quad B_r S_r &+ \quad \psi_r \end{aligned} \tag{10}$$

where $B_r = \sum_{i=1}^{d(r)-1} W^{(i)} / \sum_{i=1}^{d(r)} W^{(i)}$, $E[\psi_r] = 0$ and $Var(\psi_r) = W^{(d(r))} B_r$. This provides a basis for parents to collect information from their children. Starting from initial estimates for $\{W^{(\ell)}(0)\}$, V, and $\{\beta^{(d(r))}(0)\}$, the EM algorithm uses these in the Kalman filtering and smoothing steps, then recomputes these variance and covariate components, and repeats the process until convergence. Specifically, at step $t+1$, it first computes the expected log-likelihood of the conditional distribution of all the state variables $\{S_r\}$ given the current estimates of all variance and covariate components $\{W^{(\ell)}(t)\}$, $V(t)$, $\{\beta_\ell(t)\}$ and the data $\{y_r\}$. This is called the E-step, and uses the posterior distributions of the state variables from the Kalman filtering and smoothing steps described above. This is followed by the M-step, where we find the parameters $\{W^{(\ell)}(t+1)\}$, $V(t+1)$ and $\{\beta_\ell(t+1)\}$ that maximize the conditional distribution of $\{S_r\}$. These new estimates are now used at the next timestep. Thus, the Kalman filtering and smoothing steps are used in the "inner loop" of the EM algorithm.

The complexity of the Kalman filtering and smoothing steps is linear in the number of regions both in terms of computing time and memory requirements. On all our experiments, EM converged in less than 25 iterations. This demonstrates the scalability of our method.

4. EXPERIMENTS

We performed experiments on a large corpus of real (page, ad) data obtained as a snapshot from a set of selected servers of an active content match system. We will call this dataset *DFULL*. In Section 4.1, we describe our experimental setup and the characteristics of *DFULL*. We then verify the correctness of the first-stage algorithm for imputing impression volumes in Section 4.2, and demonstrate the effectiveness of our tree-structured Markov model in Section 4.3.

4.1 Data

For the time period considered, we obtained approximately 503 million impressions. Out of a total of approximately 6 million pages, 32,000 were in the clicked pool. From the sampled non-clicked pool, about 13,000 crawlable pages were classified into the page hierarchy. We used the same hierarchy for both ads and pages. Our hierarchy consists of 7 levels, of which the top 3 (other than the root) were considered for our analysis. They contain 20, 223 and 972 nodes respectively. Level-1 regions with less than 50 clicks were removed along with their subtrees, leaving us with a final hierarchy of 18, 177 and 860 nodes at levels 1, 2 and 3 respectively. The data shows significant sparseness in click volumes, especially at finer resolutions. In fact, about 76% and 95% of regions at levels 2 and 3 had no clicks. The maximum likelihood estimator predicts zero CTR for all zero-click regions, rendering it useless for modeling rare rates. We show that our tree-structured model provides informative estimates of CTRs for these zero-click regions by "borrowing" strength from their ancestors in the region tree.

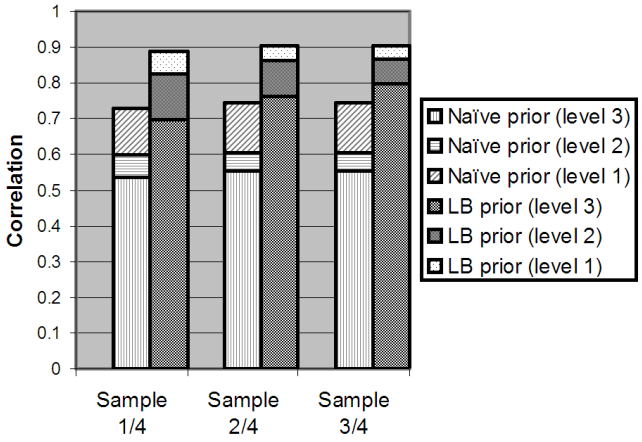


Figure 5: ***Correlation of imputed impressions with lower-bounds from* DFULL*: Each bar gives the correlation value for level* 3*, and the increase in correlation when we look at levels* 2 *and* 1 *successively. For each pair of bars, the left bar corresponds to the independence prior and the right bar to the lower-bound prior. The lower-bound prior leads to much better correlation.***

4.2 Imputing Impression Volumes

We validate the accuracy of our first-stage imputation scheme through three sets that are obtained by considering a nested sequence of sampled non-clicked pages (with sampling fractions $1/4$, $2/4$ and $3/4$) from *DFULL*. Correlation coefficients between imputed impressions $\hat{N}_r$ in each sample and lower-bounds lb_r in *DFULL* provide a measure of agreement. The rationale is as follows: when all pages are crawled, lb_r in *DFULL* is exactly the truth; when *DFULL* is only a larger validation set, the region lower-bounds are good approximations to the truth. Hence, strong correlations of the sub-samples with lower bounds in *DFULL* provide evidence of a good imputation algorithm. Figure 5 shows the correlation between $\hat{N}_r$ from the samples and lb_r from *DFULL*[1]. We also compare imputation schemes with two different priors $\{x_r(0)\}$: the lower-bound prior (LB) which assumes $x_r(0) \propto lb_r$ and the independence prior with $x_r(0) \propto 1$. Our findings are summarized below.

- *Superiority of lower-bound prior:* For all levels of the region tree, and all sample sizes, imputations based on the lower-bound prior outperform those with the independence prior. Thus, distributing excess impressions after incorporating the interactions between page and ad nodes gives better performance.

- *Monotonicity of correlations with depth:* Correlations decrease with finer resolutions due to increase in data sparsity.

- *Monotonicity of correlations with sample size:* As sample size is increased, the correlation under the lower-bound prior increases for level 3 but is relatively stable for levels 1 and 2. Thus, smaller sample sizes already contain enough data to estimate impression volumes reliably at upper levels of the tree, and increasing sample size improves performance primarily at finer resolutions.

[1]Logarithms are used to tame the skew on the original scale but all results on the original scale were qualitatively similar

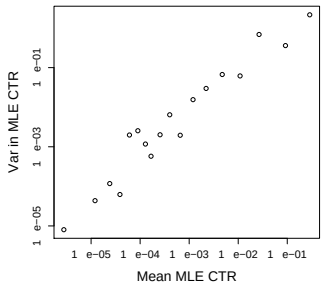


Figure 6: ***Variance vs. mean for MLE CTRs.***

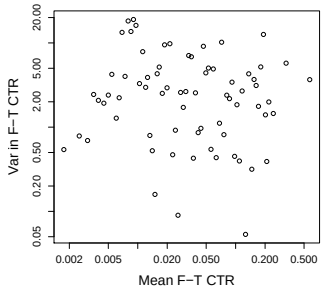


Figure 7: ***Variance vs. mean for the F-T transform.***

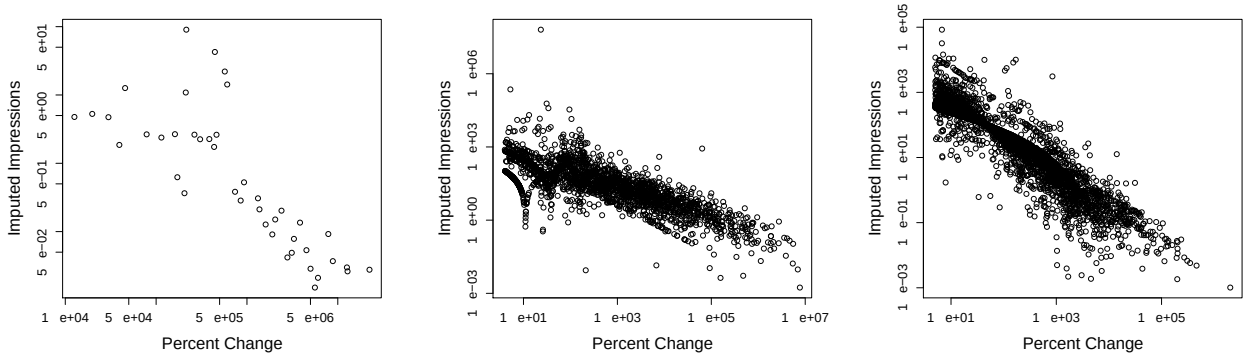
- *Strong correlations even under small sample sizes:* High correlation values provide evidence that accurate results can be obtained without crawling a large set of pages; this supports our sampling-based approach. Note that, as discussed above, accuracy and hence the desired sampling fraction are dependent on the resolution at which inference is sought.

4.3 CTR Estimation

As discussed in Section 3.2, we use the Freeman-Tukey (F-T) transformation to model our data. Figure 6 is obtained by binning the maximum likelihood estimates of CTRs with bin sizes of 100, and plotting $\sqrt{\hat{N}_r}$ times the variance in each bin against the mean in the bin. The variance clearly increases with the mean. Figure 7 shows the same plot for F-T transformed data, and no such trend is visible. This demonstrates the variance stabilization aspect of the F-T transform on our data.

Before fitting the tree-structured Markov model, we conduct an exploratory analysis to investigate evidence of sibling correlations in CTR values. We used two statistics: (1) "Moran's I," which is used as a measure of correlation among nearby regions in spatial statistics [8], and (2) F-ratio, which is used in analysis of variance (ANOVA) to compare the mean squared error among sibling groups to the mean squared error within sibling groups. Positive correlations are indicated by $I > 0$ and $F > 0$, with $I = 1$ indicating maximum homogeneity among siblings. In Table 2, both Moran's I and F-ratio show evidence of moderate global correlations (with statistical significance) in levels 2 and 3 with the effect being more pronounced at level 2. Thus, sibling regions are indeed correlated.

Next, we discuss results on our tree-structured Markov model. Our analysis was performed on a sample (henceforth *DSAMPLE*) of *DFULL* formed by sampling two-thirds of the pages from *DFULL*. We fit our model to *DSAMPLE* and validate its predictions on *DFULL*. As discussed in Section 3.2, the smoothing achieved by our model depends on the ratios $W_r/V_r = \hat{N}_r W^{(\bar{d}(r))}/V$, with lower val-



(a) Level 1 (b) Level 2 (c) Level 3

Figure 8: ***Plots comparing the percent change between the MLE CTR and the CTR found by the model fitting.***

Level		$\ell = 2$	$\ell = 3$
Moran's I	$E[I^{(\ell)}]$	0.44	0.25
	$Var(I^{(\ell)})$	0.0004	0.0004
ANOVA F-Ratio		5.77	3.77

Table 2: ***Moran's I and ANOVA scores show moderate correlations among siblings.***

	median$(\hat{N}_r) \cdot W^{(\ell)}/V$		
	$\ell = 1$	$\ell = 2$	$\ell = 3$
DSAMPLE	3002.6	6.6	.19

Table 3: ***Smoothing by level, given in terms of median***$(\hat{N}_r) \cdot W^{(\ell)}/V$***. Higher values indicate less smoothing***

ues indicating more smoothing. Table 3 provides an estimate of median$(\hat{N}_r)W^{(\ell)}/V$ for each level, and Figure 8 shows, for each level, the percentage change of the estimated values from the raw observed values. There is almost no smoothing at level 1 due to abundant data. Smoothing increases at finer resolutions as indicated by decreasing values of median$(\hat{N}_r)W^{(\ell)}/V$. Also, the amount of smoothing reduces with increasing values of $\hat{N}_r$. All these results are expected and intuitive, and serve as sanity checks on the model's performance.

We next demonstrate that CTR estimates at the finest resolution (level 3), where there is extreme data sparseness, benefit from using prior knowledge available at coarser resolutions (levels 2 and 1) through our tree-structured Markov model (*TS* hereafter). To establish a baseline, we consider two competing models. The first, called Level-Mean (*LM*), performs smoothing similar to *TS* except that the S_r values are drawn from a prior centered around 0 instead of being centered around the parent $S_{pa(r)}$. This model "shrinks" CTR estimates towards the mean CTR at level 3 and does not exploit information from coarser resolutions. The second is No-Shrinkage (*NS*), a naive model which assigns a CTR estimate of $1/\hat{N}_r$ to every region with zero clicks at level 3. Unless otherwise mentioned, all our validation is reported at level 3, and all CTR estimates are plotted on the square-root scale for exposition purposes.

For validation, we consider the set of regions $R \subseteq \mathcal{Z}^{(3)}$ with zero clicks in *DSAMPLE*. Let $R_0 \subseteq R$ be the set of regions which also had zero clicks in *DFULL*, and $R_{>0} = R \setminus R_0$ be the regions with at least one click. CTR estimates are derived from the model for each region in R, and then tested against labeled data in *DFULL*; higher scores for regions in $R_{>0}$ relative to regions in R_0 indicates better model performance, especially for predicting rare events which are the main focus of this paper.

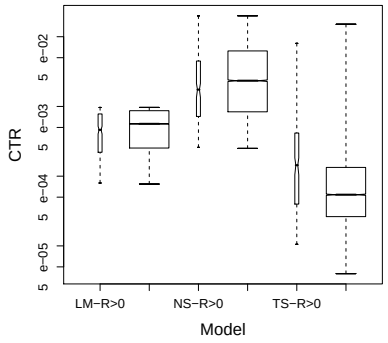
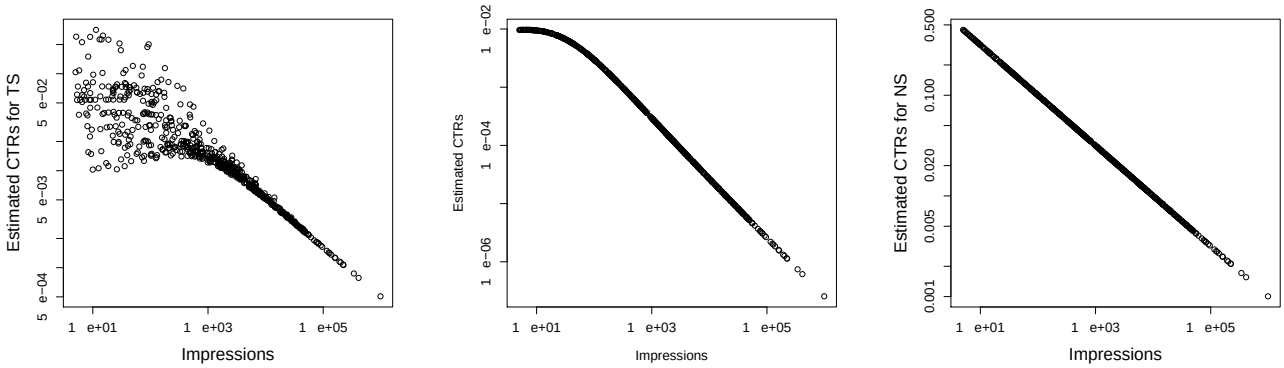


Figure 10: ***CTR estimates from regions in*** R_0 ***and*** $R_{>0}$ ***for*** **LM, NS*, and* TS.**

Before providing a comparative analysis of the three models, we provide additional insights on how smoothing occurs for *TS* and *LM*. Figure 9 shows a plot of estimated CTRs in R versus $\hat{N}_r$. For values of $\hat{N}_r$ approximately above 400, the CTR estimates are almost dictated by $\hat{N}_r$ and hence not interesting. However, the CTRs estimates from *TS* for $\hat{N}_r < 400$ are remarkably different from those obtained through *LM*. While the latter still maintains the trend of being mostly determined by $\hat{N}_r$, estimates from *TS* show a lot of variability. This occurs because in the absence of click information at level 3 and small values of $\hat{N}_r$, *TS* learns the CTRs by using information available at coarser resolutions. This is exactly what the tree-structured Markov model is designed to do: transmit information to sparse regions by learning patterns at coarser resolutions. The transmission is determined by the hierarchical structure imposed on the regions by the page and ad hierarchies, which are assumed to group regions that are similar and expected to have homogeneous CTRs. This provides evidence in favor of our hypothesis that the pre-existing hierarchies created using domain knowledge can enhance CTR predictions. Indeed, restricting ourselves to regions of R with $\hat{N}_r < 400$, Figure 10 shows the distribution



(a) *TS* (a) *LM* (a) *NS*

Figure 9: ***Estimated CTRs vs. imputed impressions for regions in*** R ***with zero clicks in*** **DSAMPLE,** ***for*** **TS, LM,** ***and*** **NS.**

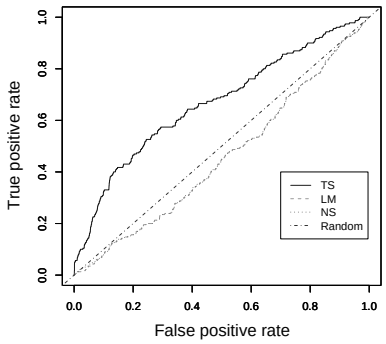


Figure 11: ***ROC plot for*** **TS, LM, NS,** ***and a random predictor that knows the proportion of*** $|R_{>0}|$ ***to*** $|R|$**. LM** ***and*** **NS** ***are almost indistinguishable.***

of CTRs for each of *TS*, *LM* and *NS* in both $R_{>0}$ and R_0. A good model should predict higher CTRs for regions from $R_{>0}$ compared to R_0. Only *TS* shows this discriminatory power, with strong statistical significance (t-statistic value=6.7, p-value=0). Corresponding results for *LM* and *NS* are weak; t-value=-2.3 (p-value=0.02) and t-value=2.03 (p-value=0.04) respectively, and have low statistical significance

Figure 11 shows ROC curves obtained using the model estimates for R and the true labels from *DFULL*. Better performance is indicated by larger area between the curve traced by the model estimate and the straight line passing through the origin. The straight line corresponds to a random classifier which labels an example as positive with probability equal to the proportion of positives in the training set; this baseline is stringent in our context since an estimate of the proportion of $|R_{>0}|$ to $|R|$ is not available from the training data. *TS* is superior to all three: *LM*, *NS* and the random classifier (even without knowing the probability of the positive class).

Overall, these results show the discriminatory power of our tree-structured Markov model, and its ability to predict CTRs even for extremely rare events (in fact, events which never even occur in the training data).

5. RELATED WORK

Multi-resolution modeling has recently become popular in the engineering and statistics literature, especially in the context of time series and spatial statistics. A rich literature in spatial statistics, known as the modifiable areal unit problem (MAUP), is related to our problem. The term MAUP was first coined by geographers Openshaw and Taylor [12] and refers to change in statistical inference based on the spatial resolution at which the data is analyzed. Two ways in which MAUP manifests itself is in the "aggregation" effect where inference changes with more aggregation and "zoning" effect where statistical inference depends on the resolution of spatial units that are considered for analysis. However, in our case we assume the taxonomies provide us with the resolutions at which it is interesting to analyze the data. We are more closely related to recent work on multi-scale tree models where nodes in the tree correspond to spatial units at different resolutions. Data are observed on some nodes and the goal is to predict at other nodes (see [5, 7] and references therein). Application of such models for estimating rates in large scale web applications is novel and to the best of our knowledge not considered before.

Imputing missing values is well known in statistics. Data imputation to adjust bias due to non-response in surveys is routine [3]. The key in any imputation scheme is the use of an appropriate model which captures the underlying mechanism appropriately. For instance, [4] discuss a novel application in ecology whereby missing pixel values are imputed using an Ising model that captures the spatial structure present in the data. The model-based procedure proposed in this paper for imputing impression volume is related to methods described in [10].

6. CONCLUSIONS

We present a method to estimate the rates of rare events at multiple resolutions in large-scale web applications. This is a challenging problem: the feature spaces are high-dimensional, the data is extremely sparse, with few impressions compared to the size of the feature space and even fewer clicks, and the scale and constant evolution of the Web render the collection of complete data expensive.

We combat these problems by considering the data at multiple resolutions using pre-existing taxonomies. Our contributions are threefold: First, we present a sampling scheme which reduces variability in CTR estimation by sampling only on pages from the non-clicked pool. Second, we present an algorithm to impute impression volumes at all levels of the taxonomy, using the sampled data and constraints induced by the taxonomies. Third, we propose a

method for simultaneously estimating CTRs at all resolutions. The method performs smoothing by using the more reliable estimates at coarser resolutions to aid inference at finer resolutions where the data sparseness problem is most extreme.

We show the effectiveness of our model on a large real-world dataset where, at the finest resolution, 95% of the regions have zero clicks. We show that even under such sparsity, our model can discriminate between regions which truly have negligible click rates from those which might get more clicks with more impressions. This could be extremely useful for online explore/exploit algorithms which, instead of having to explore each region in round-robin fashion, can now quickly home in on regions where new impressions have the best chances of generating clicks.

7. REFERENCES

[1] A.Dempster, N.Laird, and D.Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–39, 1977.

[2] G. Box and G.M.Jenkins. *Time series analysis:forecasting and control*. Holden-Day, San Franscisco, 1976.

[3] D.B.Rubin. *Multiple Imputation for Nonresponse in Surveys*. Wiley, New York, US, 2004.

[4] D.K.Agarwal, J. Silander, A.E.Gelfand, R.E.Dewar, and J. Mickelson. Tropical deforestation in madagascar: Analysis using hierarchical, spatially explicit, bayesian regression models. *Ecological Modelling*, 185(1):105–131, 2005.

[5] H.C.Huang and N.Cressie. Multiscale graphical modeling in space:applications to command and control. In *Proceedings of the Spatial Statistics Workshop, New York:Springer Lecture Notes in Statistics, Springer Verlag Publishers*, 2000.

[6] J.N.Darroch and D.Ratcliff. Generalized iterative scaling for log-linear models. *The Annals of Mathematical Statistics*, 43(5):1470–1480, 1972.

[7] K.C.Chou, A.S.Willsky, and R.Nikoukhah. Multiscale systems, kalman filters, and riccati equations. *IEEE Transactions on Automatic Control*, 39:479–492, 1994.

[8] N.Cressie. *Statistics for Spatial Data*. John Wiley, New York, 1990.

[9] N.V.Chawla, K.W.Bowyer, L.O.Hall, and W.P.Kegelmeyer. Smote: Synthetic minority oversampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002.

[10] P.Li, T.Hastie, and K.Church. A sketch algorithm for estimating two-way and multi-way associations. *Computational Linguistics(to appear)*, 2007.

[11] S.Hill, D.Agarwal, R.Bell, and C.Volinsky. Building an effective representation for dynamic graphs. *Journal of Computational and Graphical Statistics*, 15:584–608, 2006.

[12] S.Openshaw and P.Taylor. A million or so correlation coefficients. In *In N.Wrigley (Ed.), Statistical Methods in the Spatial Sciences, London*, pages 127–144, 1979.

[13] S.Pandey, D.Agarwal, D.Chakrabarti, and V.Josifovski. Bandits for taxonomies: A model based approach. In *Siam International Conference on Data Mining, Minnesota(to appear)*, 2007.

[14] C. Wang, P. Zhang, R. Choi, and M. D. Eredita. Understanding consumers attitude toward advertising. pages 1143–1148, 2002.

Appendix

Filtering: Define, for all $r \in \mathcal{Z}$, the following quantities:

$$e_r = y_r - u_r^T \hat{\beta}^{(d(r))} \quad ; \quad B_r = \frac{\sum_{i=1}^{d(r)-1} W^{(i)}}{\sum_{i=1}^{d(r)} W^{(i)}}$$

$$\sigma_r = \sum_{i=1}^{d(r)} W^{(i)} \quad ; \quad R_r = B_r W_r = B_r W^{(d(r))}$$

For the leaf regions $r \in \mathcal{Z}^{(L)}$, compute:

$$\hat{S}_{r|r} = \sigma_r e_r/(\sigma_r + V_r) \quad ; \quad \Gamma_{r|r} = \sigma_r V_r/(\sigma_r + V_r)$$

For non-leaf nodes $r \in \mathcal{Z} \setminus \mathcal{Z}^{(L)}$, let k_r denote the number of children regions under r, and let $c_i(r)$ denote the i^{th} such child. Then, compute:

$$\begin{aligned}
\hat{S}_{r|c_i(r)} &= B_{c_i(r)} \hat{S}_{c_i(r)|c_i(r)} \\
\Gamma_{r|c_i(r)} &= B_{c_i(r)} \Gamma_{c_i(r)|c_i(r)} B_{c_i(r)} + R_{c_i(r)} \\
\hat{S}^*_{r|r} &= \Gamma^*_{r|r} \left(\sum_{i=1}^{k_r} \Gamma^{-1}_{r|c_i(r)} \hat{S}_{r|c_i(r)} \right) \\
\Gamma^*_{r|r} &= \left\{ \Sigma_r^{-1} + \sum_{i=1}^{k_r} \left(\Gamma^{-1}_{r|c_i(r)} - \Sigma_r^{-1} \right) \right\}^{-1} \\
\hat{S}_{r|r} &= \Gamma_{r|r} \left(V_r^{-1} e_r + (\Gamma^*_{r|r})^{-1} \hat{S}^*_{r|r} \right) \\
\Gamma_{r|r} &= \Gamma^*_{r|r} - \Gamma^*_{r|r} (\Gamma^*_{r|r} + V_r)^{-1} \Gamma^*_{r|r}
\end{aligned}$$

Smoothing: Set the values $\hat{S}_r = \hat{S}_{r|r}$ and $\Gamma_r = \Gamma_{r|r}$ for all $r \in \mathcal{Z}^{(1)}$.

For all other levels $r \in \mathcal{Z} \setminus \mathcal{Z}^{(1)}$, compute:

$$\begin{aligned}
\hat{S}_r &= \hat{S}_{r|r} + \Gamma_{r|r} B_r \Gamma^{-1}_{pa(r)|r} \left(\hat{S}_{pa(r)} - \hat{S}_{pa(r)|r} \right) \\
\Gamma_r &= \Gamma_{r|r} + \Gamma_{r|r} B_r^2 \Gamma^{-1}_{pa(r)|r} \left(\Gamma_{pa(r)} - \Gamma_{pa(r)|r} \right) \Gamma^{-1}_{pa(r)|r} \Gamma_{r|r} \\
\Gamma_{r|pa(r)} &= \Gamma_{r|r} B_r \Gamma^{-1}_{pa(r)|r} \Gamma_{pa(r)}
\end{aligned}$$

Expectation Maximization: Define the following:

$$\begin{aligned}
e_r(t) &= y_r - u_r^T \hat{\beta}^{(d(r))}(t) \\
Q^{(\ell)}(t+1) &= \frac{\sum_{r \in \mathcal{Z}^{(\ell)}} \left(\Gamma_r + (\hat{S}_r - e_r^t)^2 \right) \hat{N}_r}{|\mathcal{Z}^{(\ell)}|}
\end{aligned}$$

Then, compute:

$$\begin{aligned}
V(t+1) &= \frac{\sum_\ell |\mathcal{Z}^{(\ell)}| \cdot Q^{(\ell)}(t+1)}{\sum_\ell |\mathcal{Z}^{(\ell)}|} \\
W^{(\ell)}(t+1) &= \frac{\sum_{r \in \mathcal{Z}^{(\ell)}} \left(\Gamma_r + \Gamma_{pa(r)} - 2\Gamma_{r,pa(r)} + (\hat{S}_r - \hat{S}_{pa(r)})^2 \right)}{|\mathcal{Z}^{(\ell)}|}
\end{aligned}$$

The value of $\hat{\beta}^{(\ell)}(t+1)$ at each level ℓ is obtained by performing a weighted least squares at level ℓ with $V(t+1)$ as estimate of V.