Privacy Preserving Frequency Capping in Internet Banner Advertising

Ayman Farahat
Addkick
Cupertino CA 94116
ayman@addkick.com

ABSTRACT

We describe an optimize-and-dispatch approach for delivering pay-per-impression advertisements in online advertising. The platform provider for an advertising network commits to showing advertisers' banner ads while capping the number of advertising message shown to a unique user as the user transitions through the network. The traditional approach for enforcing frequency caps has been to use cross-site cookies to track users. However,cross-site cookies and other tracking mechanisms can infringe on the user privacy. In this paper, we propose a novel linear programming approach that decides when to show an ad to the user based solely on the page currently viewed by the users. We show that the frequency caps are fulfilled in expectation. We show the efficacy of that approach using simulation results.

Categories and Subject Descriptors: G.3 Mathematics of Computing: Probability and Statistics

General Terms: Algorithms.

Keywords: User Model, Markov Chain

1. INTRODUCTION

Currently, advertising in the online world employs number of pricing models including cost-per-click (CPC) and cost-per-impression (CPM). While CPM or banner advertising can be implemented with an auction, in some cases brand advertisers might prefer to enter into a long-term contract with the platform provider For example, a movie advertiser might wish to guarantee a minimum number of ad views to coincide with an upcoming theater release. In order to insure that the advertising message resaches a wide range of unique users and to avoid banner burn out, the advertisers require that the maximum number of impression shown to a unique user in a fixed time period not exceed a fixed frequency cap.

In particular, we take the point of view of the advertising network that has made a number of CPM commitments with an advertiser. Each commitment specifies a price that will be paid per impression[1] and a daily frequency cap. The advertising network can show the impressions on any of the sites in network. The user's behavior on the network is assumed to follow a certain user model [4]. The user is assumed to transition from one section of the advertising network to another based on a random yet know probability transition matrix.

The traditional approach to frequency capping is to use cross-site cookies to track users through the web properties where the advertising network is serving ads. The cookies are used to keep a count of the number of ads the user has seen. When the user has reached the maximum daily caps, no further ads are shown. However, there has been growing concern over the privacy issues associated with tracking the user across multiple sites. Furthermore,up to 33 % of the users delete their cookies making cookie based approach unreliable [3].

We propose a novel algorithm that can be used to insure that the frequency caps are fulfilled in expectation. The approach is based on formulating a linear optimization program that maximizes the expected number of ads seen by the user subject to the frequency caps constraints. The solution to the linear program gives a set of probabilistic weights used by the ad server to decide whether to serve the ad when a user arrives at a specific web page.

2. RELATED WORK

Various authors have investigated the use of Markov chains to model user behavior on the web . Despande and Karypis [4] evaluate the use of Markov models to predict user activity during web session. As pointed out by [4] higher order Markov models tend to give higher predictive accuracy at the expense of increased model complexity. In this paper, we will use a first order Markov models that have been found to give adequate performance without increasing the model complexity.

In the area of privacy Canny [2] gives an algorithm whereby a community of users can compute a public aggreagte of their data that does not expose individual users data. The aggregate allows personalized recommendations to be computed by members of the community, or by outsiders. We follow a similar direction where we use aggregate information about users profile to help guide each users individual experience.

3. USER MODEL

A number of studies have shown that the user browsing behavior can be modelled using a first order Markov model. The user first arrives at the site and lands on any one of sections with a fixed probability vector $[t_{s,1}, t_{s,2} \ldots p_{s,n}] = \mathbf{p_s}$ Depending on the initial landing page, the user can transition to any of other web sections, remain on the same

[1] In this paper, we will use "impressions", "inventory", and "visits" (as in, website visits) interchangeably.

WWW 2009, April 20–24, 2009, Madrid, Spain.
ACM 978-1-60558-487-4/09/04.

section or exit the web site. Let $p_{i,j}$ denote the probability of making a transition from page "i" to page "j" where $j \in [1, 2, \ldots n, t]$ where page "t" is the terminal page. The transition matrix can be computed from panel data [3]. To illustrate the point. we consider a small but representative transition matrix.

$$\mathbf{P} = \begin{bmatrix} 0.0 & 0.5 & 0.2 & 0.1 & 0.1 & 0.0 \\ 0.0 & 0.3 & 0.2 & 0.1 & 0.0 & 0.4 \\ 0.0 & 0.1 & 0.2 & 0.2 & 0.2 & 0.3 \\ 0.0 & 0.2 & 0.2 & 0.1 & 0.2 & 0.2 \\ 0.0 & 0.0 & 0.3 & 0.3 & 0.2 & 0.2 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 1.0 \end{bmatrix} \tag{1}$$

The first state "$0''$ is the entry state and represents a new user arriving at the network. State "$5''$ is the exit state and represnets the user leaving the network. Note that the exit state is absorbing, once the user arrives at that state they can not transition to another state. It is therefore instructive to rewrite the matrix $\mathbf{P}$ as

$$\mathbf{P} = \begin{bmatrix} \mathbf{Q} & \mathbf{B} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \tag{2}$$

Following [1] , we have the fundamental matrix of the Markov Chain $M = (\mathbf{I} - \mathbf{Q})^{-1}$ and the expected number of views from the start state before reaching the absorbing state is given by

$$\mathbf{t} = M\mathbf{c} \tag{3}$$

where $\mathbf{c}$ is a column vector of all ones. A typical strategy used by ad servers is the *the greedy strategy.* In the greedy strategy, a user is always shown the banner unless they had already exceeded the frequency cap in which case they are not shown any ad. While the greedy strategy is intuitively simple, it has the major draw back that number of views is controlled by the the Markov chain governing the user behavior as described in Equation 3 (unless the advertising network is using cross-site cookies) In our model, at each section "i " of the network, the user can be shown the banner of interest with probability "q_i". Let N_i denote the total number of times a user who has just arrived at section "i" is shown the banner of interest before they leave the network. Let the vector $\mathbf{q} = [q_1, q_2, \ldots q_n]$ denote the banner display probability distribution.

$$N_i = q_i + \sum_j p_{i,j} N_j \tag{4}$$

Equation 4 gives a recursive formula for computing the expected number of banners seen by a user who has just arrived at section "i" . This number is equal to the probability of seeing a banner on section "i" , q_i plus the expected number of banners at each of the subsequent pages. The initial conditions are given by $N_0 = p_{1,s}N_1 + p_{2,s}N_s + \ldots p_{n,s}N_n$. The platform designer can adjust the serving weights $\mathbf{q}$ in order to maximize a utility function. In a frequency capped setting, the utility is such that the expected number of impressions N_0 should be as close as possible to frequency caps N_{cap} without actually exceeding that bound.

Table 1: LP Performance

num visit	greedy		lp	
num visit	$E(ad)$	$p(viol))$	$E(ad)$	$p(viol))$
3.16	3.16	0.313	3.0	0.26
6.0	6.2	0.623	3.0	0.30
10	10	0.73	3	0.32

$$\begin{aligned} & \max_{\mathbf{q},\mathbf{N}} \quad N_0 \\ \text{subject to} & \\ N_0 &\leq N_{caps} \\ N_0 &= p_{1,s}N_1 + p_{2,s}N_s + \ldots p_{n,s}N_n \\ N_1 &= q_1 + \sum_j p_{1,j}N_j \\ N_n &= q_n + \sum_j p_{n,j}N_j \\ q_i &\leq 1 \forall i \in [1, 2, \ldots N] \end{aligned} \tag{5}$$

Equation 5 gives us linear program that can be solved to get the optimal value of the display probabilities.

4. RESULTS

We simulated the policy allocation based on our algorithm. In each case, we simulated 600,000 users visting a network that is showing banner ads. The banner ads have a frequency cap of 3. The transition matrix was motivated by the matrix in Equation 1. However, we changed the entries such that the expected number of page visits as computed by equation 3 assumed different values. We compared the expected number of ads seen by a user, a $E(ad)$ and the probability of violating the frequency caps $p(viol)$ using our proposed algorithm and greedy algorithm. Table 1 shows that the LP based serving policy ensures that expected ad views meets the frequency caps and that probability of any user violating the caps is reduced.

5. CONCLUSION AND FUTURE WORK

We have presented a linear programming approach for selecting serving weights in an ad server We showed that the frequency caps are fulfilled in expectation. In future work, we will address the problem of jointly enforcing the expected number of ad views and the probability of any user violating the frequency caps. We will also address the issue of maximizing the number of clicks subject to the frequency caps and advertising budget.

6. REFERENCES

[1] P. Bremaud. *Markov Chains.* Springer Verlag, 1996.

[2] J. Canny. Collabrative filtering with privacy via factor analysis. In *ACM SIGIR Special Interest Group on Information Retrieval.*

[3] ComScore. How on-line advertising works. Technical report, 2008.

[4] M. Despande and G. Karypis. Selective markov models for predicting web page access. *ACM Transactions on Internet Technology*, 2004.