Maximizing Ad Revenue by Optimal Scheduling of Web Advertising

Mr.J.Ganesh M.E
Assistant Professor, Department of Information Technology, Dr.Sivanthi Aditanar College of Engineering
Tiruchendur TamilNadu,India

Abstract

WWW is very popular today. So many advertisers place their advertisements in web site in order to maximize the revenue. So we need an algorithm that must be optimal to place the ads. So we propose a model known as Hybrid model, where price is a function of 1) number of time the ad is exposed and 2) number of times the ad is clicked. Using this Hybrid model, I proposed two versions of solution. First one is a static where ads statically placed for some periodic time. It doesn't consider the user click behavior. Second one is dynamic where the schedule of ads is changed based in individual user click behavior. To obtain the second solution, we always maintain a parameter known as click probability. The user click behavior during a visit is observed and this information is exploited in scheduling. But in the static version, click events are not observed, and scheduling decisions are therefore made based upon the total expected click probability. As a result, a schedule that adapts to the user click behavior consistently outperforms the static solution.

I. INTRODUCTION

Nowadays most people uses internet. The internet usage is increased 50% every year. Given the large number if internet users, the web are fast becoming an attractive medium for advertising. So several web sites have now begun to depend heavily on the revenue generated by the advertisements displayed on the site. Here, we are studying an ad known as Banner ads which are a small graphic image that is linked to target web page. Many types of banner ads with different sizes and shapes are used in web ads. They are usually appear on the

1. Left side
2. Right side
3. Top
4. Bottom

of a screen as a distinct clickable image.

Generally, a web site displays a specific sequence of ads to each user during his visit to the web site. Ads compete for exposure during each time interval which may be 30 seconds or 60 seconds. Duration of user visits on a web site is called as planning horizon. A scheduling problem that arises is one of choosing subset of ads for each time interval within the planning horizon. The space for displaying ads in a time interval is referred to as a slot. The goal of the ad scheduling problem is to select a set of ads for each slot such that the total revenue over the planning horizon is maximized. Commonly used ad pricing models are namely

1. Cost Per Thousand Impression Model (CPM model)
2. Click Through Model (CTM model)
3. Hybrid model (HM model)

Now we will see about the above pricing model. In the CPM model, advertiser pays the amount to the web site owners based on the number of times the ad is displayed on the web site in a single day. It doesn't think about user visit on the ads. In the CTM model, Payment to the owner is calculated by the number of times the ad is clicked upon in the day. In the Hybrid model, Payment amount is calculated based on a combination of the number of impressions and the number of clicks on the ad. So, CPM favors web site owners because there is no risk to him whether the ad is clicked or not, he will get the amount. But CTM favors advertiser because, if someone clicks his ad then he will give amount to web site owners. Otherwise he doesn't pay amount to the web site owners. Hybrid model shares risk between both the models. I am proposing this hybrid model here. Hybrid model simply tells that space utilization objective is not equivalent to that of maximizing the revenue.

II. ASSUMPTIONS AND APPROACHES

Our model is based on the observations made by Baker [2],where it is shown that the repeated exposure of an ad during a user's visit has a negative liner effect and a positive quadratic effect on the click rate. Here, the magnitude of the negative coefficient of the linear term is significantly larger than the positive coefficient of the quadratic term. Thus, the positive quadratic effect begins to dominate only after a certain number of exposures of an ad. There are three problem constraints namely

1. Size constraints
2. Exposure constraints
3. Pair wise ad constraints

Size constraints specify that the set of ads selected for a slot fit in the slot.Exposure constraints simply tells that for each slot, at most one copy of any ad is selected for display. Pair wise constraints have two things: a) inclusion constraints and b) exclusion constraints. Inclusion constraints require that the first ad in the pair be exposed in a slot only if the second ad in the pair is exposed in that slot. For example, Computer hardware and software, Real estate and Bank ads. An exclusion constraint imposes that whenever one of the ads in the pair is exposed, the other ad in the exclusion pair cannot be displayed.

The probability of a click resulting from the *kth* exposure of an ad during a user's visit depends on two effects:

1. Exposure effect
2. Reclick effect

- ***Exposure effect.***
  Differential Impact of each successive ad exposure during a user's visit is initially negative and nonlinear but becomes positive later at higher levels of an ad exposure. Due to the exposure effect, the click probability in the *kth* exposure of an ad A_i is

$$C_{ik} = -a_{i1}k + b_{i1}k^2 + a_{i0}$$

Where a_{i0}, a_{i1}, and b_{i1} are positive constants with $a_{i1} >> b_{i1}$.

- ***Reclick effect.***
  A user clicking on an ad during s visit increases her probability of clicking an that ad in future exposures during the same visit. [4]. Thus, if an ad has been clicked upon, the reclick effect increases the click probability by an amount p in all subsequent exposures of the ad during the current visit.

These two effects generally capture the user behavior on a web site. The reclick effect has been attributed to a consumer's interest in an ad. So, for static version, click events are not observed, and therefore scheduling decisions are made based upon the total expected click probability. In dynamic version, click events are observed, and that information is included in the scheduling decisions.

In Static version, total expected click probability T_{ik} for the *kth* exposure of ad A_i is as follows:

For 1^{st} exposure, $T_{i1} = C_{i1}$,
Where C_{ik} is the click probability in the *kth* exposure of ad A_i due to the exposure effect.

For 2^{nd} exposure, $T_{i2} = T_{i1}p + C_{i2}$,

$$T_{i2} = T_{i1}(C_{i2} + p) + (1 - T_{i1})\, C_{i2}$$

In general, for $k >= 2$,

$$T_{ik} = \{ 1 - \prod_{L=1}^{k-1} (1 - C_{il})\}p + C_{ik}$$

In Dynamic version, the probability of a click in the *kth* exposure of an ad A_i is given by

$$T_{ik} = C_{ik} + p^1$$

Where $p^1 = 0$ if ad A_i has not been clicked upon thus for during the visit(exposures 1 to k-1) , and $p^1 = p$ otherwise.

In both the static and dynamic versions, the expected revenue per unit size from the kth exposure of ad A_i is given by

$$R_{ik} = a_{i2} + b_{i2}\, T_{ik}$$

Where a_{i2} and b_{i2} are positive constants.

Here, two approaches are used to implement this model. Using a linear integer programming model, we can solve the static version. For this model, I propose a heuristic solution using Static Method which uses a Timestamp Model where all ads are displayed in a round robin way with a fixed time. For dynamic model, I am using a dynamic one known as Minimum Support Threshold algorithm namely Apriori Algorithm. Computational results from this dynamic version demonstrate the importance of the observing and exploiting the click events in constructing the schedule. Generally the dynamic version outperforms the static version. The following diagram is the Architecture diagram of my work.

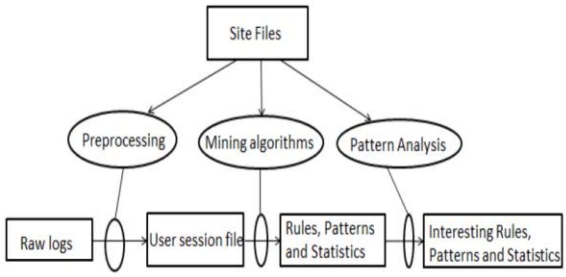


Fig 1

III. PROBLEM AND SOLUTION

Consider a set of n ads $A = \{ A_{1,......}A_n\}$ competing for space in a planning horizon that is divided into N time intervals or slots.

A Feasible schedule is a placement of a subset $A_j^1 \subset A$ of ads in slot j such that the following condition is satisfied:

a) For j= 1…..N, the sum of ad sizes assigned to slot j must not exceed S.

To illustrate, see the following example in fig1a. It has two feasible solutions that satisfy the above condition. They are represented by fig 1b and 1c.

Example: An example that illustrating the possible placements of ads.

Problem Data

Ad A_i	A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8
S_i	1	1	1	2	4	4	5	5

Fig 2a

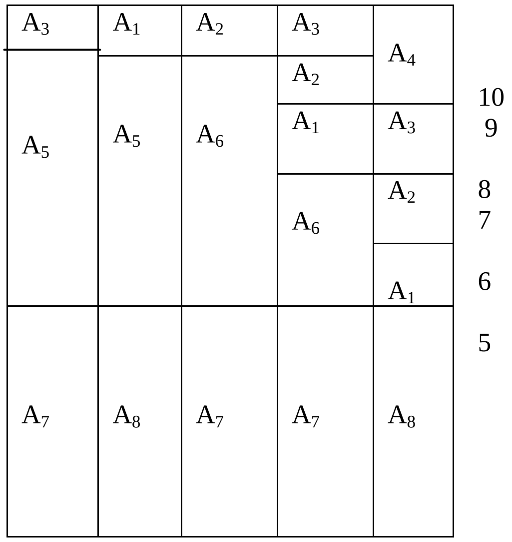
Here,
A_i - The ads are represented by $A_1, \ldots\ldots\ldots\ldots A_8$
S_i - Height of a slot.

A) Solution One

Slot 1 Slot 2 Slot 3 Slot 4 Slot 5

Fig 2b

B) Solution Two



Slot 1 Slot 2 Slot 3 Slot 4 Slot 5

Fig 2c

The following elements are basically used to find maximum amount that the web site owner actually get from advertisers. Now, we will represent the notations

N: Number of slots
n: Number of ads
S: Height of a slot.Constraint

IV. STATIC METHOD (TIME STAMP MODEL)

The basic idea behind this model is to display the ad in a round robin fashion where each ad has to be displayed in a regular interval with equal timestamp. For a problem solving more than one slot, such a problem can be solved successively for each slot. We can first determine the optimal allocation of ads in the first slot. Then, using the result of this allocation, he expected revenue of the ads displayed in the first slot can be updated, and the optimal allocation for the second slot can be obtained. The allocation for the second slot then sets up the problem for the third slot and so on.

Let m_{ij} be the expected revenue from assigning ad A_i to slot j. $X_{ij} = 1$, if the ad A_i is scheduled in slot j, 0 otherwise. We can find the maximum revenue by the modified knapsack problem for slot j by the following way.

$$MKP(j) = \text{Maximize } \sum_{i=1} m_{ij}X_{ij}$$

We can define the algorithm as follows:

Step 1: Define $m_{i1} = r_{i1}$ where $i=1,2\ldots.,n$.
Set $j=1$.
Step 2: Solve the problem MKP (j). Let X_{ij}, $i=1,2\ldots\ldots.n$ be an optimal solution for MKP(j). Set $j = j+1$.
Step 3: If $j \leq N$, update the values of m_{ij}, $i=1,2\ldots..n$, as follows: $m_{ij} = r_{ik}$, where

$$k= \sum_{q-1}^{j-1} x_{iq} + 1,$$ go to Step 2; otherwise, terminates.

V. DYNAMIC METHOD (APRIORI ALGORITHM)

The main drawback of the static Implementation is that it does not take advantage of a user's click behavior during a visit. Here, we propose an algorithm that is dynamic one which aims at exploiting this behavior. We use a fixed positive probability p to capture the reclick effect.

Dynamic begins by solving the static problem for N slots. From the corresponding solution, we use the allocation of ads in the first slot. Then, the actual click events for these ads in the first slots are observed, an allocation for the second slot is obtained by combing the old slot and the reclick of the old ads with the dynamic algorithm of Apriori which is based on the Minimum Support Threshold factor.

A. Apriori Dynamic Algorithm

The input of this algorithm is a **transaction database which is a data of reclick of ads** and a parameter called **minsup** that is a value in **[0,1].** The output of Apriori is a set of frequent patterns of the ads that is clicked. A **frequent pattern** is a pattern such that its **support** is higher or equal to **minsup**. The **support of a pattern** (also called "frequency") is the number of transactions that contains the pattern divided by the total number of transactions in the database. A key problem for algorithms like **Apriori** is how to choose a minsup value to find interesting patterns. There is no really easy way to determine the best minsup threshold. Usually, it is done by trial and error. But here we are taking from the Static method output as minsup

From this, We can find the maximum revenue by the modified algorithm for slot i by the following way.

Step 1: Solve the static problem for N slots. Based on the solution, schedule the ads in the first slot. Set q=1.
Step 2 : Observe actual click events for the ads scheduled in slot q=1.
Step 3: Calculate the updated values of u_{ik} for each ad $A_i \in A$ and each exposure k, where $M_{iq} < k \leq (N - M_{iq})$. Here,

$$M_{iq} = \sum_{h=1}^{q} X_{ih}$$

Step 4: Using $R_{ik} = a_{i2} + b_{i2} T_{ik}$ and the values of u_{ik}, calculate the updated values of $R_{ik.}$

VI. RESULTS

These are screenshots where I have did my work. Fig 3 simply gives login page of the site

Fig 3

Fig 4 actually displays the static model where the ads are displayed in a round robin way, here each ad is displayed with a constant time

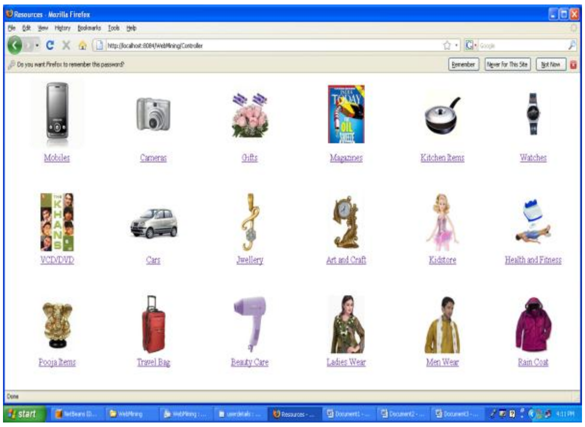


Fig 4

Fig 5 actually displays the Dynamic model where the ads are displayed based of the Apriori model, here each ad is displayed with a different time Slot which means some ads displayed more and some ads are less based on the Reclick effect

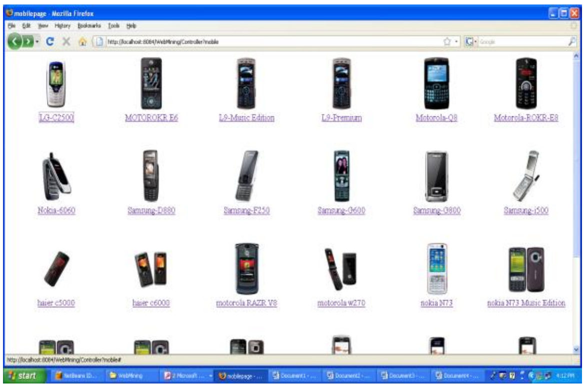


Fig 5

VII. CONCLUSIONS AND FUTURE ENHANCEMENT

Hybrid model for internet advertising is to improve the maximizing revenue of a web site. Here, we consider two versions of the problem namely, static and dynamic. Static version can be easily done by integer program, but some constraints make the problem as NP – hard. In dynamic version, the user click is observed and exploited in scheduling decisions. To solve dynamic version, we propose a look-Ahead algorithm. Look ahead algorithm is more appropriate for online ad scheduling. The dynamic algorithm is found to consistently outperform the static version for a range of parameter values.

This study that focused on optimizing web site revenue by using a hybrid pricing model is the first one. Although hybrid pricing models are gaining popularity and make good business sense, there is no result and framework that is available to analyze their impact on the total welfare of the advertising community. So, we can provide a framework and prove that hybrid pricing improves the total welfare of the advertising community.

REFERENCES

[1] M.Adler,P.B.Gibbons, and Y.Matias, “Scheduling Space-Sharing for Internet Advertising” J.Scheduling, vol.5. no.2, pp.103-119, 2002.

[2] S.Baker, “The Online Ad Surge: Brand Advertising Online Has Taken Off – And It Is shaking Up Madision Ave.,”business week”, p.76, 22 Nov.2004

[3] Dr. J. Sathish Reddy, V.Manishvarma,"Design of Adaptive Neural Network Algorithm for Superior Job-Shop Scheduling",International Journal of Mechanical Engineering (SSRG – IJME),Volume 3 Issue 3 – March 2016

[4] A.Caprara,H.Kellerer, and U.Pferschy, “ The Multiple Subset Sum Problem,” SIAM J. Scheduling, vol.11,no.2,pp.308-319,2000.

[5] P.Chatterjee,D.L.Hoffman, and T.P.Novak, “ Modeling the Clickstream: Implications for Web-Based Advertising Efforts,” sloan Center for Internet Retailing, Vanderbilit.Univ., http://elab.vanderbilit.edu/research_papers.htm,May 1998.

[6] P.Ganeshan,D.prasanna,"An intelligent scheduling for Network Traffic Management System in congestion Control using GA",SSRG International Journal of Electronics and Communication Engineering (SSRG-IJECE) – volume 1 Issue 1–Feb 2014.

[7] M.R Garey and D.S.Jhonson,Computers and Intractability: A Guide to the Thoery of NP-Completeness,freeman,1979.

[8] S.Kumar,V.S.Jacob, and C.Sriskandarajah, “ Scheduling Advertisements on a Web Page to Maximize Revenue”, European J. Operational Research , vol.173,pp.1067-1089,2006.

[9] S.Kumar, Milind dawande, and Vijay S.Mookerjee, “Optimal Scheduling and Placement of Internet Banner Advertisements”,IEEE Knowledge and Data Engineering , vol.19,no.11 Nov,2007.

[10] M.McCandless, “Web Advertising”, IEEE Intelligent systems, May-June 1998.

[11] S.Menon and A.Amiri,”Scheduling Banner Advertisments on the Web,” Informs J.Computing, vol.16, no.1, pp.95-105, Winter 2004.