Received December 4, 2019, accepted January 19, 2020, date of publication January 30, 2020, date of current version July 28, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2970463

An Effective Budget Management Framework for Real-Time Bidding in Online Advertising

MENGJUAN LIU, WEI YUE, LIZHOU QIU, AND JIAXING LI
Department of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China

Corresponding author: Mengjuan Liu (mjliu@uestc.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61202445 and Grant 61472064.

ABSTRACT Real-time bidding (RTB) has achieved great success and significantly improved the efficiency and transparency of online advertising. It allows advertisers to purchase ad impressions via auctions. Advertisers who adopt RTB always seek an optimal strategy of budget spending to reach as a wider range of audiences with a more sustainable impact as possible. Traditional bidding strategies, such as fixed bid and performance-based bid, easily lead to being either too aggressive (budgets wiped out too fast) or conservative (budget surplus exists at the end with low clicks), due to lack of optimal budget management. In this paper, we study the optimization of budget efficiency under the smooth delivery constraint for display advertising. We model the problem as a multi-constrained budget allocation optimization and use a heuristic algorithm to solve an approximate optimal budget allocation. The key to the solution is to determine the bidding function for each time slot. Here, we propose a piecewise bidding strategy to filter out the low-quality impressions, where each time slot has its own predicted click-through rate (pCTR) threshold – only when the pCTR of an impression is not lower than the threshold, will the campaign participate in the bidding. However, determining the pCTR threshold is challenging due to market uncertainty, we tackle this problem by modeling the distributions of pCTRs and market prices in each time slot. On this basis, we derive an optimal bidding function for each time slot to make the bid price more adaptable to its available budget. We conduct our experiments on a public real-world dataset and the results show that our proposed method performs the best in terms of various standard metrics (e.g., the number of clicks, cost-per-click) under a given budget comparing to the state-of-the-art baselines.

INDEX TERMS Real-time bidding, demand-side platform, budget management, bid optimization.

I. INTRODUCTION

In recent years, online advertising has developed into a multi-billion dollar industry [1]. As one of the most exciting advances in online advertising, real-time bidding (RTB) has received increasing attention, since it improves the efficiency and transparency of the display advertising ecosystem [2]. In the traditional ad delivery way, the advertisers directly buy guaranteed ad display opportunities from publishers (such as websites or mobile apps) through private contracts, where publishers (sellers) usually have partial information about the market demand of their ad impressions from historical transactions and advertisers (buyers) do not know the buying price of each other. As a result, the traditional way has some faultiness in terms of market efficiency and transparency. Different from the direct buys, RTB enables publishers to sell the individual ad impression via hosting a real-time auction, which is generally regarded as a fair and transparent way for advertisers and publishers to agree with a price quickly, whilst achieving the best possible sales outcome. Also, RTB facilitates advertisers to evaluate each ad impression and bid for it, thus more efficient budget utilization can be obtained by spending on the impressions with more positive user responses.

The typical business process of ad delivery in RTB [3] is illustrated in Fig. 1. When a user loads to browse a page of a publisher, the script for the ad slot embedded on the page will initiate a bid request for the impression to the supply-side platform (SSP). After receiving the request, SSP initiates an ad bid request to the ad exchange (ADX) with the user's cookie and the context information. Then, ADX publishes the bid request to the connected demand-side platforms (DSPs). Each DSP obtains user information from the data management platform (DMP) and starts an auction among ad campaigns that match the impression according to their targeting rules. The winner within each DSP enters the second-round auction in the ADX. The final winner will be selected at ADX according to the generalized second

The associate editor coordinating the review of this manuscript and approving it for publication was Meng-Lin Ku.

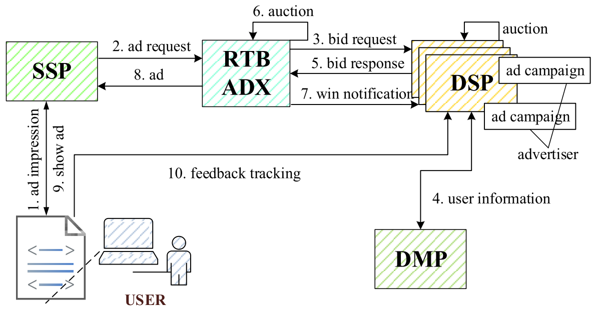


FIGURE 1. The general process of an ad delivery in RTB.

pricing (GSP) mechanism [4], that is, the actual cost of this impression is the second-highest bidding price that ADX received, also called as the market price. The winning notice is then sent to the advertiser and the winner's ad will be fed back to the SSP, and displayed to the user on the page of the publisher. In practice, the entire process needs to be completed within 100 milliseconds [14]. Later, the DSP that the winner belongs to will track the user's behavior (e.g., click, conversion), and optimize the bidding strategy based on the user response. For each day, such a request-bidding-feedback loop occurs billions of times for an ordinary RTB platform [24].

In RTB, DSP plays a critical role as the agent for advertisers and helps them manage the budget of every ad campaign. It also assists advertisers in making intelligent bidding decisions for every ad impression under specific campaign targets. Ideally, by optimizing the bidding strategy, DSP can support each advertiser to achieve the optimal utilization of budget, such as maximizing ad clicks or minimizing cost-per-click (CPC) given the fixed budget [5], [6]. Unfortunately, due to the highly dynamic and unpredictable nature of the RTB market, ad impressions and market competitions in different ad delivery periods may differ greatly. Therefore, the optimal bidding strategy learned on the historical data does not guarantee the optimal use of the total budget in the new delivery period, mainly because of the common issue existing in most bidding strategies, ignoring the budget management or smoothness of spending. That is, most of the bidding strategies may be either too aggressive, which results in that advertisers' budgets are wiped out too fast, and consequently missing the opportunities of capturing high-quality impressions in subsequent time slots, or too conservative, which ends up with the fact that budget is not fully used and remains after the entire ad delivery period, and potentially losing a number of impressions that may lead to clicks. Therefore, advertisers in RTB prefer to spend their budgets smoothly in the entire ad delivery period to reach the optimal utilization of the budget [7].

To achieve such a goal, several budget allocation algorithms are proposed. In general, they divide an ad delivery period (typically, a day) into a sequence of time slots and allocate some amount of budget for each time slot. During the ad delivery period, the spending in each time slot cannot exceed its allocated budget. However, most budget allocation algorithms are not optimal because they do not consider the quality of ad impressions (usually measured by the predicted click-through rates) in each time slot, as well as ignoring the variance of market prices in different time slots [8]. Typically, both time-based and traffic-based allocations have such shortcomings. In recent studies, the reinforcement learning (RL) framework is used to learn the optimal bidding strategy with a limited budget [9]. Different with the bidding strategies based on static optimization [5], [10], RL-based schemes decide the bid price for an impression, depending on not only the impression's value to the campaign but also the impacts of this bid on the future profits, so that the budget can be dynamically allocated across all the available impressions in the whole delivery period. Unfortunately, the performance of the RL-based bidding strategy is still unsatisfactory, although it overcomes the dynamics of the RTB environment in a sense. Therefore, in this paper we still derive the optimal bidding functions based on static optimization and control the spending rate through the allocated budget for each time slot.

On the other hand, we observe that most of the impressions with low pCTRs also have very low market prices by analyzing the iPinYou dataset, a public one used by many RTB papers [5], [9]. The situation leads to much money being wasted on those impressions with low pCTRs even if the advertisers bid with low prices. To avoid this situation, we set a pCTR threshold for each time slot to filter out those invalid impressions and give up bidding for them. However, determining the pCTR threshold for each time slot is challenging due to market uncertainty.

In this paper, we propose an effective budget management framework, for DSP to optimize the bidding strategy while satisfying the smooth delivery constraint for each campaign. Our framework consists of two interdependent components, i.e., a budget allocation algorithm and a bidding strategy. The budget allocation algorithm specifies the allocated budget for each time slot and dynamically adjusts them throughout the entire ad delivery period. The bidding strategy is a piecewise function associated with a specific time slot, for helping the ad campaign in determining the bid price on each ad impression. To our knowledge, this is the first study that improves budget utilization by combining the bidding strategy into the budget allocation algorithm. The contributions of this work can be summarized as:

- We model the budget use optimization with the smooth delivery constraint as a multi-constraints optimization problem and propose a heuristic algorithm to seek the approximate optimal budget allocation for each time slot.
- To avoid wasting the budget on the impressions with low pCTRs, a piecewise bidding strategy is proposed. That is, each time slot has its own pCTR threshold – only when the pCTR of an impression is not lower than the threshold, will the campaign participate in the bidding. Specifically, we use a simple method to derive the pCTR threshold for each time slot, by modeling two distributions of number over pCTR and market price over pCTR for each time slot.

• Following the idea of piecewise bidding, we derive an optimal bidding function for each time slot mathematically, to help the ad campaign win the high-quality impressions with reasonable prices. It depends on two factors: the pCTR of an impression and the probability function of winning a given impression with a bid price; meanwhile, a spending rate related factor is introduced to adjust the bid price to make it more adaptable for the available budget.

The rest of this paper is organized as follows. The related work is introduced in Section II. In Section III we describe the budget allocation optimization problem with multi-constraints. Section IV details the proposed budget management framework. The experimental results are presented and analyzed in Section V. Finally, we conclude our work and discuss the future work in Section VI.

II. RELATED WORK

Most of the existing bidding strategies independently decide the bid price of each impression based on its value to the campaign (usually, the value is measured by the pCTR or the predicted conversion rate) and do not take into account the constraint of delivering the ads smoothly, e.g. linear bidding [11] and non-linear bidding [5], where the bidding decision is generally formulated as a static optimization problem and the optimal bidding function is derived based on the historical data. Therefore, this kind of bidding strategies easily leads to spending out the budget prematurely or having a big budget surplus in a new ad delivery period because the RTB market is constantly dynamic. To overcome this problem, some budget allocation algorithms are proposed to work together with the existing bidding strategy. A simple, yet widely used, budget allocation scheme that meets the smooth delivery constraint is uniform allocation, where the budget is uniformly split across a delivery period. However, this time-based scheme cannot effectively differentiate the impacts of time on ad traffic and impression quality, and might force to waste money on low-quality impressions.

Therefore, several traffic-based budget allocation algorithms are accordingly proposed. For example, in [8], each ad delivery period (usually a day) is divided into a sequence of time slot schedules, and the budget allocated for each time slot is proportional to the predicted number of impressions in that time slot. In [7], Lee *et al.* propose a more effective allocation strategy, where the budget of each time slot is allocated based on the ad traffic and can be adjusted dynamically according to the actual cost. Besides, they define a term, ***pacing rate***, representing the portion of impressions that the campaign would like to bid on in a time slot, thus limiting the number of bids. The experimental evaluation shows that their algorithm provides consistent improvements in terms of CPC without under-/over- pacing. In [12], the author proves mathematically that the budget allocation proportional to the ad traffic is better than the uniform pacing, by assuming that the impressions have the same quality. However, these traffic-based budget allocation algorithms may not work well since they only consider the number of impressions in each time slot, ignoring the quality of impressions as well as the market competition.

Different from the above schemes, Xu et al. propose a smart pacing control based on pCTR [13], which first divides the impressions into several groups according to their pCTRs, and then sets different pacing rates for different groups. Typically, the impressions within the same group have the same pacing rate, and the group with a higher pCTR has a higher pacing rate. The benefit of this scheme is that it takes into account the quality of an impression when deciding whether to participate in bidding for it. But it does not support budget management, so the budget efficiency has not been guaranteed. Recently, the authors in [14] analyze that there are three main challenges when the advertisers optimizing their bidding strategies in RTB, namely (i) estimating pCTR of the ad impression, (ii) forecasting the market value of the given ad impression, and (iii) deciding the optimal bid for the given auction based on the first two. Furthermore, they point out the three challenges are strongly correlated and dealing with any individual problem independently may not be globally optimal. So they propose a comprehensive bid framework, which consists of three optimizers dealing with each challenge above, and as a whole, jointly optimizes these three parts. Unfortunately, the joint optimization scheme still regards the bidding decision as a static optimization problem, without supporting budget management.

As cutting-edge technology, reinforcement learning (RL) is introduced into the bidding decision in RTB to overcome the dynamic changes of the auction environment. Typically, Cai *et al.* [9] propose a model-based RL framework (called RLB) to sequentially generate the optimal bid price for each impression, where the bidding problem is taken as a markov decision process (MDP) and a dynamic programming algorithm is used to resolve the MDP. Specifically, at each time step (triggered by a bid request arriving), the bidding agent first observes a state (about the current RTB environment) and selects a bid action from the action space under the state, where the action is the bid price for the auctioned impression. However, such a model-based RL method requires the explicit state transition probability matrix, which is difficult to represent due to the huge computational cost in the real-world. As an improvement, the authors in [6], [15] seek to solve the MDP by using a model-free RL algorithm, where the budget-constrained bidding problem is formulated as a bidding parameter control problem based on the linear bidding function and a value-based model-free RL algorithm is leveraged to solve it. Unfortunately, both model-based and model-free bidding strategies, the performance is unsatisfactory. As a comparison, we evaluate RLB in our experiments and the results illustrate that its bidding performance is worse than that of budget allocation algorithms. Additionally, the value-based model-free RL algorithms, such as Deep Q Network (DQN) [15], may have convergence issues in practice and all have shown unable to converge to any policy for both simple MPD and simple function approximator [16].

Our budget management framework draws ideas from [7], [8], but with several improvements. Firstly, we model the budget use optimization with the smooth delivery constraint as a multi-constraint budget allocation optimization problem on time-slot level and then propose a heuristic algorithm to find the approximate optimal budget allocation for each time slot. Secondly, we propose a piecewise bidding function with the pCTR threshold, which can not only select out high-quality impressions but also combine the allocated budget with bidding decisions. Lastly, we derive an optimal bidding function for each time slot to help the ad campaign win high-quality impressions with reasonable prices.

III. PROBLEM FORMULATION

The goal of this paper is to maximize the total revenue for each ad campaign while satisfying the smooth delivery constraint under a given budget. Therefore, we model the optimal goal as a budget allocation optimization problem with multiple constraints, as shown in (1). Here, the total revenue is defined as the total number of clicks that each campaign has achieved, and B represents the total budget of an ad campaign for a delivery period. Specifically, an ad delivery period (typically, a day) is divided into T time slots, and $\{b_1^*, b_2^*, \cdots b_T^*\}$ is the optimal budget allocation for T time slots, which means the ad campaign can get the most clicks under this allocation. Here, $clicks(t)$ and $cost(t)$ are the number of clicks and the actual cost in time slot t; $\bar{b}^*$ is the average value of the optimal budget allocation and σ represents the standard deviation of the optimal budget allocation, which should be less than the threshold of smoothing constraint Θ.

$$\begin{aligned} &\underset{\{b_1^*,b_2^*,\cdots b_T^*\}}{\text{maxmize}} \sum_{t=1}^{T} clicks(t) \\ &\text{s.t. } cost(t) \leq b_t^*, \sum_{t=1}^{T} b_t^* = B, \sum_{t=1}^{T} cost(t) \leq B, \\ &\sigma = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(b_t^* - \bar{b}^*)^2} < \Theta \end{aligned} \tag{1}$$

Furthermore, based on the statistical modeling, we denote the number of clicks for each time slot as (2), where θ is the pCTR of an impression to a specific ad campaign, and $f(\theta, t)$ is the number of impressions in time slot t that are believed to have pCTR of θ, and $win(bid(\theta, t), t)$ is the winning rate for the impression with bid price $bid(\theta, t)$.

$$clicks(t) = \int_0^1 f(\theta, t) \cdot win(bid(\theta, t), t) \cdot \theta \cdot d\theta \tag{2}$$

Therefore, (1) can be rewritten as (3):

$$\begin{aligned} &\underset{\{b_1^*,b_2^*,\ldots,b_T^*\}}{\text{maxmize}} \sum_{t=1}^{T} \int_0^1 f(\theta, t) \cdot win(bid(\theta, t), t) \cdot \theta \cdot d\theta \\ &\text{s.t. } cost(t) \leq b_t^*, \sum_{t=1}^{T} b_t^* = B, \sum_{t=1}^{T} cost(t) \leq B, \\ &\sigma = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(b_t^* - \bar{b}^*)^2} < \Theta \end{aligned} \tag{3}$$

Finally, we propose a heuristic algorithm to solve (3) and get an approximate optimal budget allocation for each time slot. Before that, three functions should be determined for each time slot, i.e. $f(\theta, t)$, $win(bid(\theta, t), t)$, and $bid(\theta, t)$. In Section IV, we will detail how to solve the optimal budget allocation through two key components in our budget management framework.

IV. BUDGET MANAGEMENT FRAMEWORK

In this section, we first discuss the process of budget allocation and adjustment, and then explain how the bidding strategy works under the allocated budget and the pCTR threshold. Next, we describe the bidding strategy, the budget allocation algorithm, and the CTR estimation in detail.

Our budget management framework consists of two interdependent components: a piecewise bidding strategy and a budget allocation algorithm. Fig. 2 shows the process of budget allocation and dynamical adjustment. We first divide an ad delivery period (typically, a day) into a sequence of time slots (usually a span of 2 hours for each time slot), and then allocate the budget for each time slot by using the heuristic algorithm to solve (3). During the delivery period, the budget allocation algorithm can dynamically adjust the budgets for the subsequent time slots based on the actual cost of the finished time slots. In each time slot, we use a piecewise bidding strategy to determine the bid price for an impression. Distinguished with others, in our bidding strategy, each time slot has its own predicted click-through rate (pCTR) threshold; only when the pCTR of an impression is not lower than the threshold, will the campaign participate in the bidding. When an impression arrives at a DSP, the DSP calculates the bid prices for all eligible campaigns.

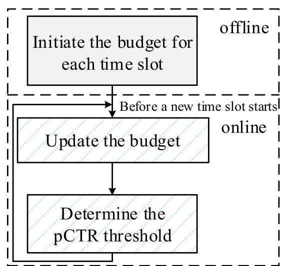


FIGURE 2. Budget allocation.

Fig. 3 shows the process of calculating the bid price. First, the bidding strategy determines whether the ad campaign has an available budget in the current time slot. If the budget is enough, it continues to estimate the pCTR of the impression and compare the pCTR with the pCTR threshold. The bid price is calculated only if the pCTR is not lower than the threshold. At last, the DSP chooses the campaign with the highest bid price to participate in the auction and return the highest bid price to the ad exchange (ADX). The ADX determines the winner and deducts the cost from the winning campaign according to the GSP (generalized second-price) mechanism.

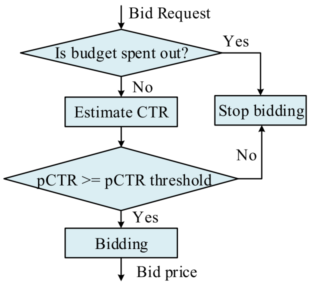


FIGURE 3. Bidding strategy.

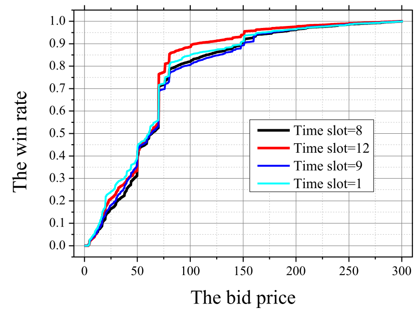


FIGURE 5. The winning rate over the bid price (advertiser ID = 1458, 2013/06/11).

A. PIECEWISE BIDDING STRATEGY

To solve (3), it is necessary to determine three functions for each time slot, i.e. the pCTR distribution function, the winning function, and the bidding function. In our budget management framework, these functions are formulated by the bidding strategy component.

Firstly, we use historical data to fit the pCTR distribution function. To do so, we construct the empirical scatter charts of pCTR distribution $f(\theta, t)$ for each time slot based on the data in the training set, and all charts exhibit a similar pattern to the F-distribution (only two example time slots are plotted in Fig. 4 for the purpose of visualization). However, it is difficult to fit the F-distribution owing to its complexity. We find that only very few points are located on the left of the peak, so we ignore these points and fit $f(\theta, t)$ only based on the points on the right of the peak (including the peak), which follows the power-law distribution. Therefore, we define $f(\theta, t)$ as (4) and the parameters (c and α) can be learned by fitting the historical data.

$$f(\theta, t) = c \cdot \theta^{-\alpha} \tag{4}$$

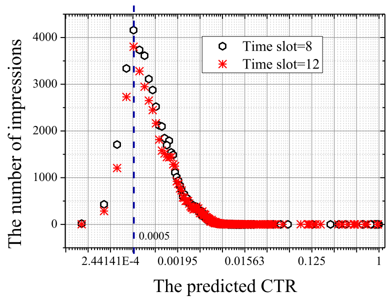


FIGURE 4. The number of impressions over the pCTR (advertiser ID = 1458, 2013/06/11).

Secondly, we estimate the winning function for each time slot by fitting the historical data. Fig. 5 shows the relationships between bid prices and winning rates in different time slots, where the differences of winning rates reflect the changes of market competitions over time. For example, when the bid price is 100 (10^{-3} Chinese FEN), the winning rates are 80.72%, 82.30%, 85.11%, and 88.67% respectively in time slot 9, 8, 1, and 12, which indicate that market competition in time slot 9 is fiercer than that in time slot 12.

Simplified, we define the winning function as (5), related to the bid price and time slot t. Here k_1 and k_2 for each time slot can be learned from the historical data.

$$win(bid(\theta, t), t) = \frac{bid(\theta, t)}{k_1 \cdot bid(\theta, t) + k_2} \tag{5}$$

Finally, we formulate the bidding function for each time slot. By analyzing the historical logs, we find that most of the impressions with low pCTRs have also low market prices. That is, the advertiser has bought many low-quality impressions even bidding at very low prices. In order to avoid wasting the budget on these low-quality impressions, a piecewise bidding strategy is designed as (6), where each time slot has its own pCTR threshold $pctr\_thresh(t)$, only when the pCTR of an impression is not lower than the threshold, will the ad campaign participate in the auction. However, there are two challenges: how to determine the pCTR threshold for each time slot and what price should offer for each high-quality impression. In the next two subsections, we will discuss them in detail.

$$bid(\theta, t) = \begin{cases} bid(\theta, t), & if\, \theta \geq pctr\_thresh(t) \\ 0, & if\, \theta < pctr\_thresh(t) \end{cases} \tag{6}$$

1) PREDICTED CTR THRESHOLD

In our budget management framework, the pCTR threshold of each time slot is crucial since it combines the piecewise bid strategy with the budget allocation algorithm. Potentially, the budget spending rate in a time slot is controlled by its pCTR threshold. In theory, the pCTR threshold relies on not only the allocated budget but also the pCTR distribution of impressions and their market prices, formulated as (7).

$$b_t^* = \int_{\theta}^{1} req(t) \cdot p(\theta, t) \cdot g(\theta, t) d\theta = \int_{\theta}^{1} f(\theta, t) \cdot g(\theta, t) d\theta \tag{7}$$

Here, $f(\theta, t)$ denotes the number of impressions in time slot t that are believed to have pCTR of θ, which can be represented by the number of impressions $req(t)$ and the probability density function of pCTR $p(\theta, t)$; $g(\theta, t)$ represents the market price of the impression with the pCTR of θ. The meaning of (7) is that the sum of the market prices of impressions whose pCTRs are no less than the threshold θ equals to b_t^* in time slot t, assuming all impressions selected by the threshold should be won.

To calculate the pCTR threshold, we need to formulate two functions $f(\theta, t)$ and $g(\theta, t)$. Since $f(\theta, t)$ has been defined by (4), we only need to formulate $g(\theta, t)$. Firstly, we analyze the relationship between the market price and the pCTR of an impression, shown as Fig. 6, where the range of pCTRs [0, 1] is equally divided into 10000 intervals and each point is the average market price of the impressions whose pCTRs belong to an interval. The scatter plots for two example time slots show no significant correlation between the market price of an impression and the pCTR. This is because the market price of an impression is mostly determined by the pCTRs to all ad campaigns (bidders) in GSP auction, rather than the pCTR to one specific campaign. Thus it is difficult to accurately fit $g(\theta, t)$ based on the historical data. But, we also observe the fact that the average market price of impressions with high pCTRs is mostly higher than that of impressions with low pCTRs in Fig. 7, where the impressions in two example time slots are arranged in ascending order of pCTR and each point represents the average market price of every 5000 adjacent impressions. Furthermore, we divide the sorted impressions in time slot 8 into two groups and find that the average market price of the first group with low pCTRs is 64.47, significantly lower than that of the second one with high pCTRs, i.e. 77.13. Thus, in this paper, we simply use the average market price of high-quality impressions instead of $g(\theta, t)$, as shown in (8), where $mprice(i)$ is the market price of impression i and I represents a set of impressions with pCTRs no less than β (hyperparameter), N is the number of impressions in I.

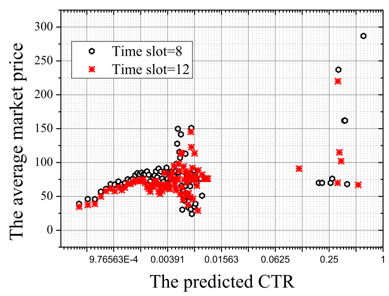


FIGURE 6. The average of the market prices over the pCTR (advertiser ID = 1458, 2013/06/11).

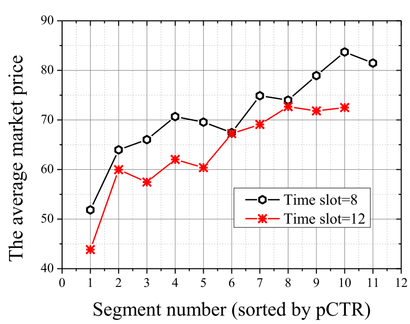


FIGURE 7. The average of the market prices over the segment number (each segment contains 5000 impressions where all impressions are sorted in ascending order according to pCTR, advertiser ID = 1458, 2013/06/11).

$$\overline{mprice} = \frac{1}{N}\sum_{i\in I} mprice(i) \tag{8}$$

Then (7) can be rewritten as (9):

$$b_t^* = \int_{\theta}^{1} c\cdot\theta^{-\alpha}\cdot\overline{mprice}\, d\theta \tag{9}$$

Solving (9) gives the pCTR threshold of time slot t:

$$pctr\_thresh(t) = \theta = \sqrt[1-\alpha]{1 - \frac{1-\alpha}{c\cdot\overline{mprice}}\cdot b_t^*} \tag{10}$$

2) OPTIMAL BIDDING FUNCTION

A bidding function refers to the logic of deciding a bid price for a given impression. In most of the bidding strategies, each campaign learns an optimal bidding function and bids for all available impressions in the whole delivery period based on this bidding function. Obviously, it is not the best way for ignoring the impacts of market competitions in different time slots on the winning rates. Therefore, in this paper, we derive the optimal bidding function for each time slot to make the bid prices more suitable for the dynamic RTB market on time-slot level.

The goal of our optimal bidding function is to maximize the number of the clicks in a time slot, as defined in (11), where θ is the pCTR of an impression. Here, the lower bound of the integral range is $pctr\_thresh(t)$, because that in the piecewise bidding strategy each campaign only bids for the impressions whose pCTRs are not below the threshold. Furthermore, we take (4) and (5) into (11) and find that $bid(\theta, t)$ is the unique variable of the objective function. Thus, (11) is a functional extremum problem [16]. We define the Lagrange function of (11) in (12), where λ is the Lagrangian multiplier.

$$\text{maxmize} \int_{pctr\_thresh(t)}^{1} f(\theta,t)\cdot win(bid(\theta,t),t)\cdot\theta\, d\theta \tag{11}$$

$$s.t. \int_{pctr\_thresh(t)}^{1} f(\theta,t)\cdot win(bid(\theta,t),t)\cdot bid(\theta,t)d\theta \le b_t^*$$

$$L(\theta, bid(\theta,t), bid'(\theta,t)) = f(\theta,t)\cdot win(bid(\theta,t))\cdot\theta - \lambda\cdot f(\theta,t)\cdot win(bid(\theta,t),t)\cdot bid(\theta,t) \tag{12}$$

Here, the Euler-Lagrange equation is given as follows.

$$\frac{\partial L}{\partial bid} - \frac{d}{d\theta}\left(\frac{\partial L}{\partial bid'}\right) = 0$$

We take (12) into Euler-Lagrange equation and get the following derivations:

$$\frac{\partial L}{\partial bid'} = 0, \quad \frac{d}{d\theta}\left(\frac{\partial L}{\partial bid'}\right) = 0,$$

since $L(\theta, bid(\theta,t), bid'(\theta,t))$ does not contain $bid'(\theta,t)$.

Therefore, we can get (13).

$$\begin{aligned}\frac{\partial L}{\partial bid} &= \frac{\partial L(\theta, bid(\theta,t), bid'(\theta,t))}{\partial bid(\theta,t)} = 0 \\ &\Rightarrow f(\theta,t)\cdot\theta\cdot\frac{\partial win(bid(\theta,t),t)}{\partial bid(\theta,t)} \\ &= \lambda\cdot f(\theta,t)\cdot win(bid(\theta,t),t) \\ &\quad + \lambda\cdot f(\theta,t)\cdot bid(\theta,t)\cdot\frac{\partial win(bid(\theta,t),t)}{\partial bid(\theta,t)} \\ &\Rightarrow \theta\cdot\frac{\partial win(bid(\theta,t),t)}{\partial bid(\theta,t)} \\ &= \lambda\cdot win(bid(\theta,t),t)+\lambda\cdot bid(\theta,t)\cdot\frac{\partial win(bid(\theta,t))}{\partial bid(\theta,t)}\end{aligned} \tag{13}$$

Taking a derivative with respect to $bid(\theta, t)$ gives:

$$\frac{\partial win(bid(\theta,t))}{\partial bid(\theta,t)} = \frac{k_2}{(k_1\cdot bid(\theta,t)+k_2)^2} \tag{14}$$

Taking (5) and (14) back into (13) gives:

$$bid(\theta,t) = \sqrt{\frac{\lambda\cdot(k_2)^2+\theta\cdot k_1\cdot k_2}{\lambda\cdot(k_1)^2}} - \frac{k_2}{k_1} \tag{15}$$

Furthermore, considering the dynamic nature of the market, we also introduce a variable factor that is related to the spending rate of the current time slot. Therefore, (15) is reformulated as follows:

$$bid(i,t) = (sp(t)/sp'(i))\cdot bid(\theta,t) \tag{16}$$

where $sp(t)$ is the ratio of the available budget of time slot t to the duration of a time slot, representing the ideal spending rate of time slot t; $sp'(i)$ is the ratio of the actual cost to the elapsed time of the current time slot, representing the instantaneous spending rate when the impression arrives. Thus, our bid price can be adjusted dynamically according to the real market competition to better adapt to the available budget in the current time slot. In Section V, our experimental results show that our bidding function can perform the best under the given budget.

B. BUDGET ALLOCATION ALGORITHM

The original idea of budget allocation is to take the daily budget as input and calculate the scheduled spending for each ad campaign. Based on the scheduled spending, each campaign can achieve the goal of smooth ad delivery throughout the lifetime. We use a heuristic algorithm to solve the multi-constrained optimal budget allocation problem shown in (3). Based on historical data, we randomly set up 5,000 budget allocation schemes that satisfying the smooth constraint and choose the budget allocation scheme with the largest number of clicks in the training set as the optimal allocation, $B^* = \{b_1^*, b_2^*, \cdots, b_T^*\}$.

In practice, for an ad campaign, there is always a difference between the actual cost and its allocated budget of a time slot; therefore, we take a dynamic sequential approach to adjust the budget for the next time slot based on the actual spending of the finished time slots, similarly to [7], [13]. The budget for the next time slot can be updated by a simple function defined in (17), where the updated budget of time slot t is b_t' and $spend(i)$ represents the actual cost of the campaign in time slot i.

$$b_t' = (b_t^*\cdot(B-\sum\nolimits_{i=1}^{t-1} spend(i)))/\sum_{i=t}^{T} b_i^* \tag{17}$$

C. CTR ESTIMATION

As mentioned above, a good budget management framework relies heavily on the performance of the CTR Estimator. There are many studies on the CTR estimation; however, designing a good CTR estimator remains a challenge, because the click event is extremely sparse in the real dataset [17]. In this paper, the focus is on the budget management, not the CTR estimation. Therefore, we empirically compare the performance of the five representative models and choose the best one as our CTR estimator. Among them, three models are based on the shallow structure, i.e. logistic regression (LR) [18], factorization machine (FM) [19], and field-aware factorization machine (FFM) [20], and two models are based on the deep neural network, i.e. factorization machine supported neural network (FNN) [21] and Product-based Neural Network (PNN) [22].

V. EXPERIMENTAL RESULTS AND DISCUSSIONS

In this section, we conduct experiments on a publicly available real-world dataset iPinYou[1] and evaluate the performance of the proposed budget management framework by comparing with several state-of-the-art baselines. Before presenting the detailed experimental results, we first briefly describe the dataset.

A. iPinYou DATASET

Our dataset is published by iPinYou, one of the leading DSP companies in the online advertising industry. It includes logs of impressions, bids, clicks, and final conversions. For each impression, the bid logs contain the information of the user. So we can train a better CTR estimator based on such rich data. In addition, the impression and click logs provide the information of bid price, paying (market) price, and user feedbacks (e.g. clicks), and whether the advertiser won the auction. More details of the dataset can be found in [23]. In our experiments, we use the data of 7 days from 2013/06/06 to 2013/06/12. The aggregated data for the first 6 days is employed as the training set, and the data on the last day is used to construct the testing set. Note that in our experiments we use the winning impressions as the received impressions, ignoring the losing impressions, because the losing impressions have no market prices and user feedbacks. Therefore, the number of impressions in our experiments is far less than the actual number in the iPinYou dataset. Besides, since the budget management of each campaign

[1] iPinYou Dataset website: http://data.computational-advertising.org/

TABLE 1. The statistics of the campaign (advertiser ID = 1458).

Date	2013/6/6	2013/6/7	2013/6/8	2013/6/9	2013/6/10	2013/6/11	2013/6/12
Cost	30,096,630	30,228,554	30,615,541	30,548,604	30,303,929	30,309,883	30,297,100
Reqs	3,250,536	3,733,421	1,390,738	1,851,641	1,551,919	1,485,384	1,437,857
Win bids	448,164	478,109	413,804	423726	434,240	437,520	447,493
Clicks	328	307	347	351	370	395	356
Win rate	13.7874%	12.8062%	29.7543%	22.8838%	27.9808%	29.4550%	31.1222%
CTR	0.0732%	0.0642%	0.0839%	0.0828%	0.0852%	0.0903%	0.0796%
CPM	67.1554	63.2252	73.9856	72.0952	69.7861	69.2766	67.7041
CPC	91758	98464	88229	87033	81903	76734	85104

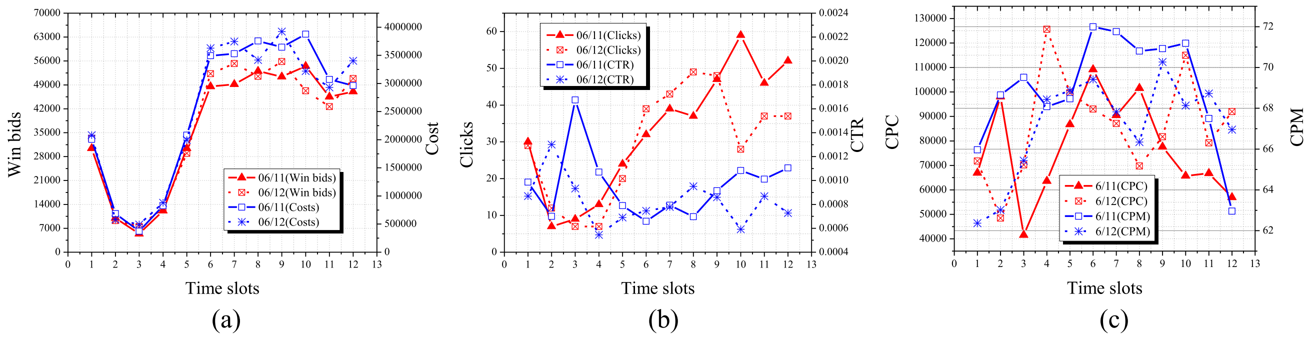


FIGURE 8. The distributions of key metrics (advertiser ID = 1458, 2013/06/11 and 2013/06/12).

is performed independently in our framework, we present the statistical and experimental results of one advertising campaign (advertiser ID = 1458) due to space limitations. We have consistent and similar results for other campaigns.

The descriptive statistics of this campaign are shown in Table 1. In the iPinYou dataset, the DSP bids for every impression with a fixed bid price of 300, and all numbers related to price use the currency of RMB, and the unit is 10^{-3} Chinese FEN. The click-through rates in all 7 days are very low, ranging from 0.0642% to 0.0903%, which will be likely to significantly reduce the accuracy of the CTR estimators. Furthermore, we observe that there is no significant change in the number of win bids (which are taken as the received impressions in our experiments) per day, which means we can estimate the key functions (such as the pCTR distribution function, the winning function and the bidding function) and metrics (such as the pCTR threshold) of each time slot in a new delivery period based on the historical data in the training set. Unfortunately, it is difficult to analyze the accurate evolving patterns in these statistics since the dataset only has records of 7 days. Actually, some key metrics such as clicks and CPC are quite different every day. Therefore, when seeking the optimal budget allocation through a heuristic algorithm, we fit the key functions and metrics of each time slot every day in the training set based on the data that really happened in that time slot. However, for the new ad delivery period (e.g. the 7th day in the testing set), the RTB market is unknown, in this paper we simply learn the key functions and metrics of each time slot on the 7th day just based on the data on the 6th day (the day before the testing day). We do this on the assumption that the closer the time, the more similar the market environment. In addition, we split a day into 12 time slots and allocate a budget for each time slot to satisfy the smooth delivery constraint.

Fig. 8 shows the real statistics of each time slot on the 6th and 7th days. We can observe that the statistics distributions of each time slot on the two days are quite different, except the metrics of win bids and cost. This is because that the RTB market (including impressions, market competitions and user feedbacks) is a constantly dynamic environment even for the same ad campaign. Specifically, Fig.8 (a) exhibits the distributions of winning impressions and costs on the two days, and the statistical results show that the winning impressions and costs in the first 6 time slots are significantly lower than those in the last 6 time slots. Especially from 2:00 a.m. to 8:00 a.m., the winning impressions and costs are very small. Fig.8 (b) focuses on the statistics of clicks and CTRs and both of them exhibit different distributions. For example, in time slot 10, the number of clicks on the 6th day is higher than that on the 7th day; while in time slot 8, the reverse is true. Also, the CPC and the CPM of each time slot are displayed in Fig.8 (c), where the CPMs (the average market prices) on the 6th day are much higher than those on the 7th day in time slot 1, 2, 3, 6, 7, 8, 9 and 10, implying that in these time slots the market competitions on the 6th day are more intense than those on the 7th day. Therefore, the method that uses the historical data on the 6th day to fit the key functions and metrics on the 7th day has some defects, which can deteriorate the performance of our budget management framework. However, it is still regarded as a feasible method due to the dynamic and unpredictable nature of the RTB market. And for comparison, we also evaluate the performance of the proposed framework under the budget of 5,000,000 by learning the key functions and metrics based on the historical data of all 6 days in the training set, where the number of clicks is 239 and the CPC is 19878 (10^{-3} Chinese FEN), worse than those of the scheme that learns the key functions and metrics on the 6th day.

B. CTR ESTIMATION

Although the CTR estimator is not the focus of this paper, a good one is critical to improving the effectiveness of budget management framework. In this experiment, we evaluate the performance of five CTR estimators – LR, FM, FFM, FFN, and PNN, by using the standard metrics: AUC value and LogLoss. The experimental results are shown in Table 2. Obviously, the FFM model has a significant advantage in predicting CTR among the five models in terms of AUC. Therefore, we use the FFM model to predict CTR for every impression in the subsequent experiments.

TABLE 2. Performance comparison of CTR estimators.

Model	LR	FM	FFM	FNN	PNN
AUC	0.8100	0.8114	0.8742	0.8117	0.8172
LogLoss(10^{-3})	4.4364	4.4103	5.4519	4.5024	4.5649

C. COMPARISON OF BIDDING STRATEGIES

In this section, we compared our bidding strategy with other four bidding strategies: fixed bidding, linear bidding [11], non-linear bidding [5], and RL-based bidding [9] in terms of various performance metrics, such as the number of clicks, CTR (click-through-rate), and CPC. In our dataset, the paying prices fluctuate between 0 and 300. Accordingly, we select a set of fixed bid prices from 5 to 300, increased one by each time. The linear bidding function is defined as

$$bid(i, t) = cpv(t) \cdot pCTR(i)/epCTR(t) \tag{18}$$

where $cpv(t)$ and $epCTR(t)$ are the historical average market price and the historical average pCTR of impressions in time slot t, and $pCTR(i)$ is the predicted CTR of impression i. In [5], the nonlinear bidding function is defined as (19), which only depends on the pCTR of the impression. Here, both c and λ are parameters and they do not change over time slots. In RL-based bidding, the bid price of each impression is determined by an optimal action-selection policy, which is learned through a model-based RL framework. Our bidding function is related to the specific time slot, depending on the winning function of each time slot. Note that we only compare the five bidding strategies, without supporting any budget management. Therefore, in this experiment, our bidding function for each time slot has not the pCTR threshold. The daily budget for the campaign is 5,000,000 (10^{-3} Chinese FEN).

$$bid(i, t) = \sqrt{\frac{c}{\lambda} pCTR(i) + c^2} - c \tag{19}$$

Firstly, we find that in all experiments with fixed bids the best performance can be obtained when bidding price is 75. Therefore, we only show the results of the experiment with the fixed bid 75. The results are shown in Table 3 and the distributions of key metrics for each time slot are shown in Fig. 9. It tells us that RL-based bidding is superior to others, which obtains the highest number of clicks and the lowest CPC. This is because that the bid decision is modeled as a sequentially dynamic interactive process in RL-based bidding, where the bid price of an impression is determined based on both the immediate and long-term future rewards. That proves the reinforcement learning is indeed beneficial to solve the optimization problem in the dynamic environment. Exclude RL-based bidding, our bidding strategy performs best. The worst bidding strategy is the fixed bid with 75, followed by the linear and non-linear bids. In fact, there is a large difference in terms of the number of clicks between the best bidding strategy (153) and the truth (356) on the testing day, since the given budget is only 16.5% of the total cost of all impressions on the testing day. Empirically, except RL-based bidding, all strategies spent out their budgets too early due to the absence of the budget management. Desirably, RL-based bidding bids for all available impressions in the whole delivery period, achieving the best performance.

TABLE 3. Performance comparison of bidding strategies.

Bidding strategy	Bids	Imps.	Clicks	CPC	CTR (%)
Fixed(75)	146,511	109,415	76	65788	0.0694
Linear	114,135	94,944	85	58823	0.0895
Non-Linear	130,805	102,245	95	52631	0.0929
RL-based	447,493	116,463	153	25331	0.1314
Ours	144,617	104,757	101	49504	0.0964

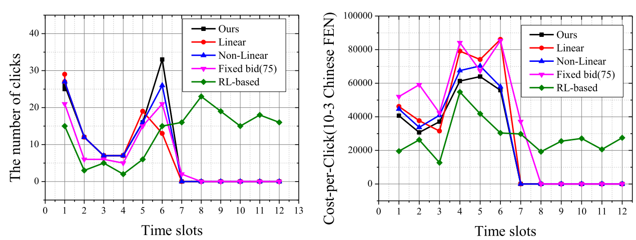


FIGURE 9. The distributions of key metrics under different bidding strategies without any budget management.

Furthermore, Table 4 summarizes the reasons for losing clicks in four bidding strategies (excluding RL-based bidding). And the results reveal that most of the missing clicks are due to the budget wiped out too early. Non-linear stopped

TABLE 4. The number of missing clicks for different reasons (R1: budget wiped out; R2: bid price bellowed the market price).

Reason	Fixed(75)	Linear	Non-Linear	Ours
R1	238	267	249	242
R2	42	4	12	13
Total	280	271	261	255

biding at time slot 6, the earliest ones; linear, fixed (75) and our strategy all stopped bidding at time slot 7; as a result, they lost all impressions that could bring clicks in the subsequent time slots.

D. COMPARISON OF BUDGET ALLOCATION ALGORITHMS

In this subsection, we validate the performance by imposing the budget allocation algorithm to the bidding strategy. Our budget allocation algorithm consists of two components: budget allocation and its dynamic adjustment. In this experiment, we introduce some representative budget allocation algorithms as baselines.

- **No Budget Allocation (NBA)**: this scheme bids for each bid request without budget allocation, also not limiting the cost in each time slot.
- **RL-based Bidding (RLB)**: this scheme bids for each bid request according to a bid price-selection policy learned by a model-based RL framework, without budget allocation.
- **Uniform Allocation (UA)**: this scheme splits the daily budget uniformly across the day, and does not support dynamic adjustment of the budget.
- **Uniform Allocation with Adjustment (UAA)**: this scheme allocates budget for each time slot uniformly, while supporting dynamic adjustment.
- **Traffic-based Allocation with Adjustment (TAA)**: in this scheme, the allocated budget is proportional to the predicted number of impressions in each time slot, and can be adjusted dynamically according to the actual cost.
- **Optimal Allocation with Adjustment (OAA)**: the budget is allocated and adjusted for each time slot according to our budget allocation algorithm.

1) PERFORMANCE COMPARISON

We first compare the performance of six algorithms under the daily budget of 5,000,000, as shown in Table 5. Then we give a detailed description of the distributions of key metrics throughout a day, which are illustrated in Fig. 10. It is noted that our piecewise bidding strategy is adopted by all schemes in this experiment, except NBA and RLB. In NBA, the bidding function for each time slot bids for an impression according to (15); and in RLB, the bid price is determined by an optimal bid price-selection policy. Therefore, neither of them has the pCTR threshold and budget constraint for each time slot.

Firstly, we observe that the performance of the four schemes with budget management (UA, UAA, TAA, and OAA) has indeed improved greatly, compared with NBA and RLB. Specifically, the budget allocation and the piecewise bidding strategy with the pCTR threshold can

TABLE 5. Performance comparison of different budget allocation algorithms (budget = 5,000,000).

Scheme	Bids	Imps.	Clicks	CPC	Smoothness
NBA	144,617	104,757	101	49504	567,116
RLB	447,493	116,463	153	25331	153,724
UA	66,207	60,702	224	17937	22,187
UAA	80,251	73,410	237	20340	69,003
TAA	76,540	70,633	241	19797	209,092
OAA	78,154	73,452	244	20184	131,702

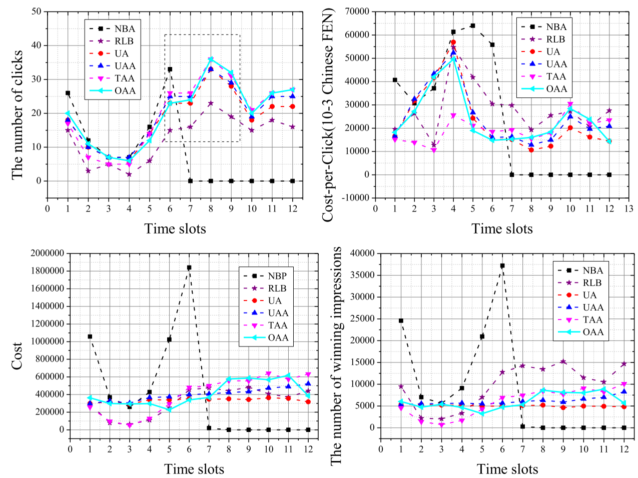


FIGURE 10. Detailed performance comparison of six budget allocation algorithms on the 7th day (budget = 5,000,000).

indeed spend money on impressions that have higher click probability, which play important roles in improving the bidding performance. Secondly, the adaptive adjustments of budgets for subsequent time slots are very small, as shown the cost curves of UA and UAA, because we introduce a factor that is related to the spending rate of the current time slot when calculating the bid price and the budgets of the most time slots are almost spent out. At last, we further observe that the number of clicks of our budget allocation is slightly higher than those of UAA and TAA. This is because that our optimal budget allocation is learned based on the historical data, which only includes logs for 6 days and is insufficient for predicting the distributions of some metrics (such as clicks and the average market prices). Additionally, the standard deviations of cost in each time slot under the six allocation schemes are also shown in Table 5, recorded as smoothness. We find that the smoothness of our optimal budget allocation is better than that of RLB, TAA, and NBA, worse than that of UA and UAA. The results illustrate that the budgets can be spent smoothly in the whole delivery period by imposing the budget allocation algorithm on the bidding strategy; meanwhile, RLB also has the ability to control the smooth use of the budget.

Furthermore, Table 6 discusses the reasons for missing the actual clicks on the testing day. Here, we ignore the statistics of RLB because there is only one reason for losing the actual clicks. That is, the bid prices of these impressions are lower than their market prices. For other four schemes

TABLE 6. The number of missing clicks for different reasons (budget = 5,000,000; R1: budget wiped out; R2: lower than the market price; R3: lower than the threshold).

Reasons	NBA	UA	UAA	TAA	OAA
R1	242	0	0	0	0
R2	13	18	21	22	11
R3	0	114	98	93	101
Total	255	132	119	115	112

with budget allocation, the primary reason is that the pCTRs of these impressions are lower than their pCTR thresholds and only a few clicks are lost because of their bidding prices lower than their market prices. We further analyze why most of the time slots have high pCTR thresholds. According to (10), the pCTR threshold of each time slot depends mainly on the allocated budget and the average market price of high-quality impressions. Here, the higher the average market price, the higher the pCTR threshold. In this experiment, we use the average market price of high-quality impressions in each time slot on the 6th day to estimate that on the 7th day. However, the estimated average values are significantly higher than their real values (as shown in Fig. 8 (c)), so the derived pCTR thresholds in most time slots are higher than the real pCTR thresholds. Also, the high pCTR thresholds result in that those schemes with budget management have not spent out their budgets, even though the budget is only 16.5% of the total cost of all impressions on the7th day.

2) PERFORMANCE UNDER DIFFERENT BUDGETS

In this experiment, we study the impact of daily budget on the performance of budget management. So we set the daily budget to 10,000,000, 33% of the total cost of all impressions in testing set.

The results are summarized in Table 7 and the distributions of key metrics throughout a day are shown in Fig. 11. We find that the numbers of clicks in all schemes increase significantly with the allocated budgets increasing; especially for NBA, the number of clicks has risen up to 56.74% of the actual clicks. Unfortunately, the CPC and the total cost of OAA are higher than those of UAA and TAA, although OAA still has the highest number of clicks. This is because that the allocated budgets for many time slots in OAA are higher than those in UAA and TAA, which lead to the pCTR thresholds of these time slots are lower. Thus, OAA can buy more invalid impressions than UAA and TAA. On the other hand, the CPC of OAA is increased by 65.24%, compared with that under the budget of 5,000,000. This is because that the pCTR threshold of each time slot decreases as the budget increases and our piecewise bidding strategy bids on more impressions with low pCTRs. In summary, under different daily budgets, our budget allocation and the piecewise bidding strategy can still work well.

TABLE 7. Performance Comparison among Different Budget Allocation Algorithms (budget = 10,000,000).

Scheme	Bids	Imps.	Clicks	CPC	CPM	Smoothness
NBA	286,836	208,782	202	49504	47.8968	722,277
RLB	447,493	189,082	210	36929	41.0152	307,671
UAA	168,076	151,465	286	33122	62.5421	221,238
TAA	164,893	148,286	285	32854	63.1459	423,119
OAA	171,738	155,267	291	33352	62.5098	338,152

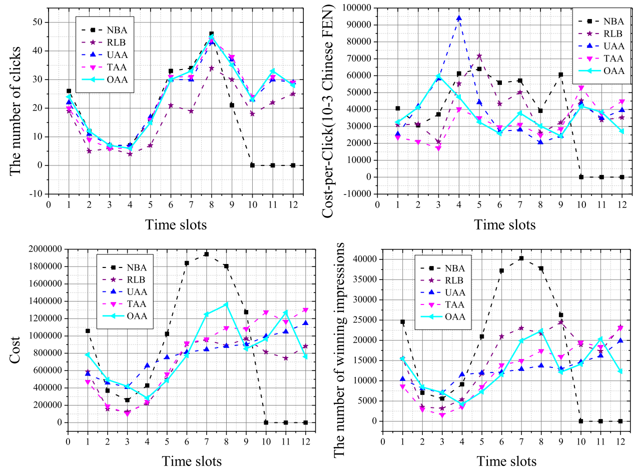


FIGURE 11. Detailed performance comparison of five budget allocation algorithms on the 7th day (budget = 10,000,000).

TABLE 8. Performance comparison under different bidding strategies (budget = 100,000,000).

Scheme	Bids	Imps.	Clicks	CTR (%)	CPC
Linear	235,694	189,360	175	0.0924	57142
Non-Linear	307,386	214,578	214	0.0997	46728
OAA*	360,552	219,446	233	0.1062	42918
OAA	171,738	155,267	291	0.1874	33352

E. DIFFERENT BIDDING STRATEGIES WITH OUR BUDGET ALLOCATION

In this section, we compare the performance of different bidding strategies under the proposed budget allocation algorithm. The daily budget is still set to 100,000,000. We take the linear, non-linear and our optimal bidding without the pCTR threshold (called as OAA*) as the baselines. The results are shown in Table 8. We can see that the performance of our piecewise bidding (OAA) is best, followed by OAA*, Non-Linear and Linear, which demonstrate that our piecewise bidding is effective since it focuses money on impressions with high pCTRs. Table 9 analyzes the reasons for missing clicks. For those bidding strategies without the pCTR thresholds that the allocated budget wiped out early is the main reason for losing clicks. The results demonstrate that

TABLE 9. The number of missing clicks for different reasons (budget = 10,000,000; R1: budget wiped out; R2: lower than the market price; R3: lower than the threshold).

Reasons	Linear	Non-Linear	OAA*	OAA
R1	170	105	61	0
R2	11	37	62	30
R3	0	0	0	35
Total	181	142	123	65

our piecewise bidding strategy with pCTR threshold is indeed effective for improving the bidding performance.

VI. CONCLUSION

In this paper, we present a budget management framework to maximize the obtained clicks of an ad campaign under budget and smooth delivery constraints. It allocates the sub-budget for each time slot while considering not only the expected number of clicks but also budget consumption speed. We design a piecewise bidding strategy with the pCTR threshold to select high-quality impressions and implement a time-varying bidding function to capture the dynamics of RTB for improving the performance. In our experiments conducted on a real-world dataset, we compare our budget allocation algorithms with state-of-the-art baselines under different budget amounts and various bidding strategies. The experimental results show that our proposed methods can effectively improve the revenue of advertisers and achieve smooth delivery. In our future work, we plan to further investigate the bidding strategies based on the machine learning models (such as [14]) and the latest reinforcement learning models (such as [6], [9]).

REFERENCES

[1] J. Xu, X. Shao, J. Ma, K. C. Lee, H. Qi, and Q. Lu, "Lift-based bidding in ad selection," in *Proc. AAAI*, Phoenix, AZ, USA, 2016, pp. 651–657.

[2] S. Yuan, J. Wang, and X. Zhao, "Real-time bidding for online advertising: Measurement and analysis," in *Proc. ACM-ADKDD*, Chicago, IL, USA, 2013, pp. 19–27.

[3] J. Wang, W. Zhang, and S. Yuan, "Display advertising with real-time bidding (RTB) and behavioural targeting," *Found. Trends Inf. Retr.*, vol. 11, nos. 4–5, pp. 267–435, 2017.

[4] Y. Yuan, "A survey on real time bidding advertising," in *Proc. IEEE-SOLI*, Qingdao, China, Oct. 2014, pp. 418–423.

[5] W. Zhang, S. Yuan, and J. Wang, "Optimal real-time bidding for display advertising," in *Proc. ACM-SIGKDD*, New York, NY, USA, 2014, pp. 1077–1086.

[6] J. Zhao, G. Qiu, and Z. Guan, "Deep reinforcement learning for sponsored search real-time bidding," in *Proc. ACM-SIGKDD*, London, U.K., 2018, pp. 1021–1030.

[7] K. C. Lee, A. Jalali, and A. Dasdan, "Real time bid optimization with smooth budget delivery in online advertising," in *Proc. ACM-ADKDD*, Chicago, IL, USA, 2013, pp. 1–9.

[8] D. Agarwal, S. Ghosh, K. Wei, and S. You, "Budget pacing for targeted online advertisements at LinkedIn," in *Proc. ACM-SIGKDD*, New York, NY, USA, 2014, pp. 1613–1619.

[9] H. Cai, K. Ren, and W. Zhang, "Real-time bidding by reinforce-ment learning in display advertising," in *Proc. ACM-CWSDM*, Cambridge, U.K., 2017, pp. 661–670.

[10] Y. Chen, P. Berkhin, and B. Anderson, "Real-time bidding algorithms for performance-based display ad allocation," in *Proc. ACM-SIGKDD*, San Diego, CA, USA, 2011, pp. 1307–1315.

[11] C. Perlich, "Bid optimizing and inventory scoring in target-ed online advertising," in *Proc. ACM-SIGKDD*, Beijing, China, 2012, pp. 804–812.

[12] J. Fernandez-Tapia, *Optimal Budget-Pacing for Real-Time Bidding*. Rochester, NY, USA: Social Science Electronic Publishing, 2015.

[13] J. Xu, K. C. Lee, W. Li, H. Qi, and Q. Lu, "Smart pacing for effective online ad campaign optimization," in *Proc. ACM-SIGKDD*, Sydney, NSW, Australia, 2015, pp. 2217–2226.

[14] K. Ren, W. Zhang, K. Chang, Y. Rong, Y. Yu, and J. Wang, "Bidding machine: Learning to bid for directly optimizing profits in display advertising," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 4, pp. 645–659, Apr. 2018, doi: 10.1109/tkde.2017.2775228.

[15] D. Wu, X. Chen, X. Yang, H. Wang, Q. Tan, X. Zhang, J. Xu, and K. Cai, "Budget constrained bidding by Model-free reinforcement learn-ing in display advertising," in *Proc. ACM-ICIKM*, New York, NY, USA, 2018, pp. 1443–1451.

[16] O. P. Agrawal, "Formulation of euler-Lagrange equations for fractional variational problems," *J. Math. Anal. Appl.*, vol. 272, no. 1, pp. 368–379, Aug. 2002, doi: 10.1016/S0022-247X.(02) 00180-4.

[17] O. Chapelle, E. Manavoglu, and R. Rosales, "Simple and scalable response prediction for display advertising," *ACM Trans. Intell. Syst. Technol.*, vol. 5, no. 4, pp. 1–34, Jan. 2014, doi: 10.1145/2532128.

[18] M. Richardson, E. Dominowska, and R. Ragno, "Predicting clicks: Estimating the click-through rate for new ads," in *Proc. WWW*, Banff, AB, Canada, 2007, pp. 521–530.

[19] S. Rendle, "Factorization machines with LibFM," *ACM Trans. Intell. Syst. Technol.*, vol. 3, no. 3, pp. 1–22, May 2012, doi: 10.1145/2168752.2168771.

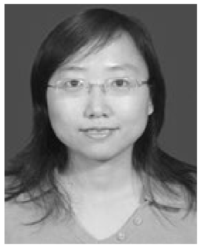
[20] Y. Juan, "Field-aware factorization machines for CTR prediction," in *Proc. ACM-CRS*, Boston, MA, USA, 2016, pp. 43–50.

[21] W. Zhang, T. Du, and J. Wang, "Deep learning over multi-field categorical data: A case study on user response prediction," in *Proc. ECIR*, Padua, Italy, 2016, pp. 45–57.

[22] Y. Qu, H. Cai, K. Ren, W. Zhang, Y. Yu, Y. Wen, and J. Wang, "Product-based neural networks for user response prediction," in *Proc. IEEE-ICDM*, Barcelona, Spain, Dec. 2016, pp. 1149–1154.

[23] H. Liao, L. Peng, Z. Liu, and X. Shen, "iPinYou global RTB bidding algorithm competition dataset," in *Proc. ACM-ADKDD*, New York, NY, USA, 2014, pp. 1–6.

[24] J. Shen, B. Orten, S. C. Geyik, D. Liu, S. Shariat, F. Bian, and A. Dasdan, "From 0.5 million to 2.5 million: Efficiently scaling up real-time bidding," in *Proc. IEEE-ICDM*, Atlantic City, NJ, USA, Nov. 2015, pp. 973–978.



MENGJUAN LIU received Ph.D. degree in communication and information system from the University of Science and Technology of China, in 2007. She is currently an Associate Professor with the Department of Information and Software Engineering, University of Electronic Science and Technology of China. Her main research interests are in the areas of computational advertising, data mining, and artificial intelligence.

WEI YUE was born in Henan, China, in 1995. He received the B.S. degree from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2017, where he is currently pursuing the M.S. degree in software engineering with the School of Information and Software Engineering. His research interests include computational advertising and machine learning.

LIZHOU QIU was born in 1995. He received the B.S. degree from the School of Computer and Communication Engineering, Changsha University of Science and Technology, China, in 2017. He is currently pursuing the master's degree in software engineering with the School of Information and Software Engineering, UESTC. His main research interests include computational advertising, recommendation systems, and data mining.

JIAXING LI was born in Sichuan, China, in 1996. He received the B.S. degree in software engineering from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2018, where he is currently pursuing the M.S. degree in software engineering with the School of Information and Software Engineering. His research interests include machine learning and computational advertising.