Optimizing the ecosystem of an internal ads platform

ABSTRACT

Many e-commerce websites have an internal ads platform that has control over both algorithmic bidding and allocation. It allows the advertisers to promote their products and yield higher revenue. Meanwhile, it helps the website to generate extra revenue from bidding and conversion of the ads. Compared to traditional ads platform modeling, modeling this type of platform requires consideration of performance of both advertisers and ad suppliers, which is the website itself. In this work, we propose a simple and scalable revenue maximization framework to solve the above problem. The model can be dynamically adjusted in an multiplicative weight update fashion to meet the goal performance specified by the platform and the advertisers. Extensive experiments have been run on a real-world dataset from an e-commerce company to validate the effectiveness of the model.

CCS CONCEPTS

• **Information systems → Sponsored search advertising**; • **Computing methodologies → Modeling methodologies**; **Modeling methodologies**;

ACM Reference Format:

. 2017. Optimizing the ecosystem of an internal ads platform. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 MOTIVATION AND INTRODUCTION

Online advertising is an important source of revenue for many Internet services ranging from e-commerce websites to social networks. It is a billion-dollar industry and is still expanding at a double digit growth rate. [1]

One category of online ads is sponsored-search ads, which usually appear on search result pages. In 2002, Google initiated the AdWords service for sponsored search ads, where advertisers bid for a key word and the winners (whose ads are shown) are determined by the Generalized Second Price Auction (GSP). Recently, with a rapid growing number of websites and advertisers, this type of auction-based bidding has a variant called real-time bidding (RTB). RTB mechanism has the following 2 features: (1) bidding price for each advertiser is submitted in real time depending on the user information (eg, search query, browsing and purchase history) and contextual information (eg, time and date, browser and geographical region) and (2) an auction is held to determine the winning advertisers and they are charged based on certain user behavior, eg: impression, click, and purchase. One merit of RTB is that it provides rich information to a large pool of advertisers and thus helps matching users and advertisers efficiently.

Three key platforms have emerged to make this micro ad market function properly at a large scale: Demand side platform (DSP), Supply side platform (SSP) and the Ad Network. DSP serves the advertisers, managing their campaigns and submitting algorithmic bidding to the Ad Network. SSP serves the publishers, managing their websites and ads allocation. Ad Network is a market maker that holds the auctions and makes buying and selling an impression (a page shown to a user) possible.

DSP and SSP have inherently different, sometimes even conflicting goals. DSP functions to lead to more clicks and conversions at lower bidding costs, while SSP functions to generate higher bidding revenue. To achieve these goals, several lines of research work have been done to model DSP and SSP separately. For DSP, one fundamental question is bid optimization. It needs to consider various practical constraints including the budget of advertisers and maximal cost an advertiser is willing to pay for each auction. Then DSP will optimize towards performance of ad campaigns. The performance is measured in different metrics: for optimizing number of clicks and conversions, see [17]; for optimizing other novel metrics used in video advertising, see [8]. Optimizing towards the total profit for DSP is considered in a recent work by Grigas et al[9]. For SSP, one central question is how to increase the total bidding revenue from the advertisers. There have been some work on determining attribution and setting the reverse price [7, 14]. For a more comprehensive survey of works on real time bidding, see [16].

In this work, we consider an end to end platform that plays the role of DSP, SSP and Ad Network. On the one hand, the platform has full control over displayed contents; on the other hand, the platform provides the opportunities for content generators to compete for a few 'ads' positions using auction. Examples for this type of platform includes an e-commerce website like Amazon that also promotes certain products, and a social network website like Facebook that promotes products and subscriptions.

This platform (1) controls algorithmic bidding for advertisers; (2) holds a real time auction for each search query and charges the advertisers based on clicks; (3) shares a percentage of attribution/purchase revenue from the ads, if it is clicked and converted. Note that the first assumption and second automatically implies the platform functions as both DSP (bids for advertisers) and SSP (provides impressions). As an SSP, it wants to maximize the total bidding revenue. However, as a DSP, it needs to take the objective of advertisers into consideration, which is more clicks and attributions at less bidding costs. It is the nature of balancing those conflicting objectives that makes optimization for this type of platform interesting. Indeed, a good balance of revenue and ads serving is necessary guarantee long term health of the platform.

We will propose a simple revenue maximization framework for

[1] https://www.iab.com/news/digital-ad-revenues-grow-19-year-year-first-half-2016/

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

ads bidding and allocation with various performance (clicks, attributions, return over investments) constraints to solve the above multi-objective balance problem. The proposed model is efficient in computation and scalable to real time bidding. Moreover, it has the flexibility to adjust to various real-time performance metrics specified by the advertisers. The model is tested on a real world dataset from an e-commerce website.

2 RELATED WORK

Recent research on RTB focuses on demand side platform (DSP), supply side platform (SSP) and Ad network. For DSPs in real time bidding, they bid based on the impression level features: user information and contextual information. After a bid is submitted, an auction is held to determine the winning ads. A generalized second price (GSP) auction is often used in practice, where the winning advertisers pays the amount of money to achieve the second lowest ranking score. Therefore, one of the key problems for DSP is bid optimization.

From a modeling perspective for DSP, GSP makes the the relation between actual payment from advertisers and their bidding price non-linear and the actual payment is dependent on other advertisers. One common approach is do probabilistic modeling of the bidding landscape, i.e, to estimate the probability of winning given a bid and other contextual information [6]. In [17],the authors use a parametrized family to approximate the landscape distribution.

Other model constants that need to be estimated include expected number of clicks and expected number of purchases because they are directly related to the the performance of the advertisers. Click through rate (CTR) and conversion rate (CVR) prediction are the key quantities. In particular, CTR is the probability an ad will be clicked given user and contextual information and CVR is the probability an ad will be converted given the same information. CTR and CVR predictions are still an active area of research. With recent advance in deep neural networks and language modeling, techniques from these fields are adapted to improve the prediction of these metrics, see [10], [5],[12]. [3] propose a generative model for CVR distribution modeling and solve it using Expectation-maximization (EM) algorithm.

With the above components, several optimization models have been proposed. They differ in optimizing different objectives and actual optimization algorithms. In [17], it proposes a functional optimization framework to maximize the total number of clicks. In [9], it proposes a 2-phase Lagrangian algorithm to maximize the total profit of DSPs. In [4], it solves the revenue maximization problem through ads allocation using linear programming (LP). The LP in their model admits integral optimal solution and has a nice primal-dual structure that automatically reveals the bidding and ranking rules for the auction.

Our proposed optimization model solves a slightly different problem in the sense that the revenue includes the shared purchase revenue besides bidding revenue. Similar to [9], we include both allocation and bidding variables. However, our model includes more constraints besides the budget constraints. For instance, we add different performance constraints of the advertisers because the platform functions as DSP and needs to guarantee the performance of the advertisers. Note that in our optimization model, we only consider first price auction to get some insight of the structural property of the model. Under proper relaxation, the problem can be solved using LP. In a similar vein as [4], we will show that the dual LP gives us a ranking function and the structural optimal bidding strategy. Furthermore, the adaptive bidding algorithm we have proposed has a natural connection with the notion of pacing. Pacing is a recent research interest among different DSPs, and its goal is to spend the budget of advertisers smoothly throughout the lifetime of the campaign [2],[11],[15]. There are two type of techniques for pacing, bid modification [1] and probabilistic throttling [2],[15]. Our structural optimal bidding strategy is very similar to bid modification.

3 MODEL AND THEORY

3.1 Notation

(1) impression level: every impression is associated with a **query** q_i, $i = 1, \dots N$, generated by a user.
(2) auction level: there are M ad campaigns, each denoted as a_j, $j = 1, \dots, M$. For each advertiser a_j, we assume that it manages one ad for simplicity, with a fixed budget B_j. After query i comes, there are online estimates of CTR and CVR of a_j with respect to the search query, denoted as $c_{i,j}$, and $C_{i,j}$ respectively.
(3) payment level: we use r to denote the shared percentage of attribution revenue from ad if it is clicked and converted. $mCPC_j$ is the maximal cost per click advertiser a_j is willing to pay.

3.2 Model Description

The model is the following: after a search query comes, each advertiser will submit a bid, and the platform will hold an auction to determine the winner for that ad slot. If the winning ad is clicked, then the advertiser will be charged the amount of money determined by the auction. Moreover, if the ad is purchased, the advertiser and the platform will share a percentage of the purchase revenue. We consider the setting where only one ad slot is allocated for each search query.

The platform will help advertisers bid algorithmically and design a ranking rule to determine the winners. Therefore, the decision variables for the model are $W_{i,j} \in \{0, 1\}$ and $B_{i,j} \geq 0$ for $1 \leq i \leq N$, $1 \leq j \leq M$, where $W_{i,j}$ is a binary variable indicating whether a_j wins the auction for search query i, and $B_{i,j}$ is the bid a_j submits for search query q_i.

The expected utility for an ad j with bid $B_{i,j}$ for search query q_i is:

$$c_{i,j}B_{i,j} + c_{i,j}C_{i,j}R_j * r$$

where the first component $c_{i,j}B_{i,j}$ is the expected bidding revenue, and the second component $c_{i,j}C_{i,j}R_j * r$ is the expected shared purchase revenue. Note that we use the following factorization for the purchase probability:

$$\begin{aligned}
&P(\text{purchase}|\text{ad } j, \text{query } i)\\
=&P(\text{purchase and click}|\text{ad } j, \text{query } i)\\
=&P(\text{purchase}|\text{click, ad } j, \text{query } i)P(\text{click}|\text{ad } j, \text{query } i)\\
=&C_{i,j}c_{i,j}
\end{aligned}$$

Here, we only model the first price revenue, and we will show later that the algorithm can actually be used for the general second price auction. The constraints we have are of the following types: (1) allocation constraint (2) budget constraint (3) bidding constraint and (4) performance constraint.

- *allocation* constraint
$$\sum_{j=1}^{M} W_{i,j} = 1 \ \ \forall i \tag{1}$$
We assume only 1 ad slot is available. This constraint can be extended to multiple winning slots by replacing 1 with the number of winning slots.
- *budget* constraint
$$\sum_{i=1}^{N} c_{i,j} W_{i,j} B_{i,j} \le B_j \ \ \forall j \tag{2}$$
It says that the expected spending of the advertiser should not exceed its budget.
- *bidding* constraint
$$B_{i,j} \le mCPC_j \ \ \forall i,j \tag{3}$$
The advertisers can specify the maximal CPC it is willing to pay for each auction.
- *performance* constraints
The performance constraints are designed by the platform. If it sets the goal for total number of clicks, then the constraint is of the following form:
$$\sum_{i=1}^{N}\sum_{j=1}^{M} W_{i,j} c_{i,j} \ge G_{\text{click}} N$$
where G_{click} is the target global click through rate. If it sets the goal for the shared purchase revenue, then it is:
$$\sum_{i=1}^{N}\sum_{j=1}^{M} W_{i,j} c_{i,j} C_{i,j} R_j * r \ge G_{\text{attribution}}$$
If it sets the goal for the global rate of return for the advertisers, then it is:
$$\sum_{i=1}^{N}\sum_{j=1}^{M} W_{i,j} c_{i,j} C_{i,j} R_j \ge G_{ROI} \sum_{i=1}^{N}\sum_{j=1}^{N} W_{i,j} c_{i,j} B_{i,j}$$

We give three examples here to show the flexibility of adding performance constraints into the model. G_{click}, $G_{\text{attribution}}$ and G_{ROI} are the hyperparameters set by the platform.
The objective is to maximize the total expected revenue:
$$\max \ \sum_{i=1}^{N}\sum_{j=1}^{M} W_{i,j} c_{i,j} (B_{i,j} + C_{i,j} * R_j * r)$$

3.3 Relaxation

In the above formulation, the products $W_{i,j}B_{i,j}$s' make the problem non-convex. We will introduce $Z_{i,j} = W_{i,j}B_{i,j}$ and relax $W_{i,j} \in [0,1]$. The relaxation for the above problem without the performance constraints is the following:
$$\begin{aligned} \max \ & \sum_{i=1}^{N}\sum_{j=1}^{M} c_{i,j} Z_{i,j} + c_{i,j} C_{i,j} W_{i,j} R_j * r \\ \text{s.t} \ & \sum_{j=1}^{M} W_{i,j} \le 1 \ \ \forall i \\ & \sum_{i=1}^{N} c_{i,j} Z_{i,j} \le B_j \ \ \forall j \\ & Z_{i,j} \le mCPC_j W_{i,j} \ \ \forall i,j \\ & W_{i,j}, Z_{i,j} \ge 0 \ \ \forall i,j \end{aligned}$$
Note that for bidding constraint, we multiply both sides by $W_{i,j}$, then $W_{i,j}B_{i,j} \le mCPC_j W_{i,j} \leftrightarrow Z_{i,j} \le mCPC_j W_{i,j}$. The relaxation problem is a linear programming (LP), and it admits efficient algorithms to solve. However, the above LP does not satisfies total uni-modular (TU) condition, and the integrality of the optimal solution is not guaranteed. Moreover, the above problem is off-line formulation, given that we already know $c_{i,j}$ and $C_{i,j}$. In reality, we really need to solve the online version of the above problem, where we know $c_{i,j}$ and $C_{i,j}$ only after the query i arrives.
This motivates us to look at the dual linear programming of the above problem. This approach has been studied in [4] for a simpler setting. Indeed, we will see shortly that the dual problem admits simple optimal solution structure and will allow us to obtain the ranking function for the auction.
Let β_is' be the dual variables for the allocation constraints, α_js' be the dual variables for the budget constraints, and $r_{i,j}$s' be the dual variables for the bidding constraints. The dual LP is the following:
$$\begin{aligned} \min \ & \sum_{i=1}^{N} \beta_i + \sum_{j=1}^{M} B_j \alpha_j \\ \text{s.t} \ & r_{i,j} + c_{i,j}\alpha_j \ge c_{i,j} \ \ \forall i,j && (4) \\ & \beta_i - r_{i,j} mCPC_j \ge c_{i,j} C_{i,j} R_j * r \ \ \forall i && (5) \\ & \alpha_j, \beta_i, r_{i,j} \ge 0 \ \ \forall i,j \end{aligned}$$
The optimal solution structure for the dual problem is very simple, summarized in the following proposition.

Proposition 3.1. *Suppose that the primal problem is feasible, then a dual optimal solution α_i^*, β_j^*, $r_{i,j}^*$s' will satisfy the following conditions:*

- $\alpha_j^* \in [0,1] \ \ \forall j$
- $r_{i,j}^* = (1-\alpha_j^*) c_{i,j} \ \ \forall i,j$
- $\beta_i^* = \max_j c_{i,j}((1-\alpha_j^*) mCPC_j + C_{i,j} R_j * r) \ \ \forall i,j$

We interpret β_i's as the shadow price for each search query, and α_j's as the cost/penalty for additional bidding. Assume that we know the optimal solution α_j^*s', then set the bid for ad j to be $(1-\alpha_j^*) mCPC_j$ for each query i, and rank the ads by their expected utility:
$$c_{i,j}((1-\alpha_j^*) mCPC_j + C_{i,j} R_j * r)$$
The winner ad j^* has the optimal value β_i^* and it will be allocated for the ad slot. For simplicity, we assume that for each search query i, there is only one winner $j^*(i)$, which means that $c_{i,j}((1 -$

$\alpha_j^*)mCPC_j + C_{i,j}R_j * r) < \beta_i^*$ for all $j \neq j^*(i)$ (If there is a tie between winning advertisers, the winning slot will be allocated to one of them with equal probability).

The total expected revenue generated by this bidding and allocation method is:

$$\sum_{i=1}^{N} c_{i,j}^*(i)((1-\alpha_{j^*(i)}^*)mCPC_{j^*(i)} + C_{i,j^*(i)}R_{j^*(i)} * r) = \sum_{i=1}^{N} \beta_i^*$$

Thus the gap from the optimal value is $\sum_{j=1}^{M} \alpha_j^* B_j$. This gap is 0 iff α_j^*s' are all equal to 0, and that will correspond to a greedy bidding model: for each search query, the platform just bids the maximal CPC each advertiser is willing to pay. It is an optimal strategy for the platform if advertisers have a sufficient amount of budget throughout a day. However, it is not necessarily optimal if the advertisers are involved in a plenty of auctions throughout a day with limited budget. Greedy implies that an advertiser is likely to bid higher in the early period of the day and do not have sufficient budget for the future auctions.

Using the optimality condition for the primal-dual solutions, we have that:

$$\beta_i^*(1 - \sum_j W_{i,j}^*) = 0 \tag{6}$$

$$W_{i,j}^*(\beta_i^* - c_{i,j}((1-\alpha_j^*)mCPC_j + C_{i,j}R_j * r)) = 0 \quad \forall i,j \tag{7}$$

By (6) and (7), we see that only the advertiser $j^*(i)$ with highest expected utility will be allocated, with $W_{i,j^*(i)} > 0$ and $W_{i,j} = 0$ for $j \neq j^*(i)$. This is our main intuition to set the bidding and allocation for advertisers as above.

In the online version of the above problem, we do not know α_j^*s apriori. However, by the primal-dual structure, we have intuition that α_js' are related to how each advertisers has consumed their budget. In particular, the complementary slackness guarantees that $\alpha_j^*(B_j - \sum_{i=1}^{N} c_{i,j}Z_{i,j}^*) = 0$ for all j for optimal primal and dual solution.

We will use an adaptive control for the α_js' to estimate the optimal α_j^*s'. Throughout a day's auction, we set a few checkpoints to update α_js' based on the planned budget spending and the actual budget spending. Let $N_0 = 0 < N_1 < N_2 < \cdots < N_T$ be the checking point, and define:

$$S_j(t) = \sum_{i=N_{t-1}}^{N_t} B_{i,j} I(clicked)$$

It is the actual budget spending of advertiser j between N_{t-1} and N_t search queries/impressions. $B_j(t)$ be the planned spending budget between N_{t-1} and N_t search queries. The updating formula for α is:

$$\alpha_j(0) = \alpha_0$$

$$\alpha_j(t+1) = \max(\alpha_j(t)\exp(\gamma(\frac{S_j(t)}{B_j(t)} - 1)), 1)$$

This type update method is also used in [4]. Interestingly, here we can connect the adaptive control with the notion of pacing in online advertising. Pacing is a technique to spend the advertisers' budget smoothly throughout the day by DSPs. Any pacing schemes can be specified by $B_j(t)$, $t = 1, \cdots, N$.

If it is uniform, then $B_j(t)$ is the same for all t. If it is traffic based, then it is correlated with the volume of traffic. We have the following key observation: when the actual spending for advertiser j exceeds the planned spending, α_j will increase, and thus the effective bidding $(1-\alpha_j)mCPC_j$will decrease in the next time period. Conversely, if the actual spending for advertiser j lags the planned spending, α_j will decrease and the effective bidding will increase in the next time period. Mehta [13] used bid modification as follows:

$$b_j^* = b_j * (1 - \exp(\frac{S_j(t)}{B_j} - 1))$$

where b_j can be interpreted as max bid for advertiser j. Note that if we set $B_j(t)$ to be the same for all t, then the proposed updating formula is very close to Mehta's modified bid. In particular,

$$\alpha_j^t \approx \alpha_0(\gamma N(\frac{S_j(t)}{B_j} - t/T))$$

3.4 Algorithm

We summarize the above procedure in the following algorithm

Data: Ads budget, maximal CPC
α_0, γ, checkpoints ;
while *not the end of day* **do**
  current query = q_i;
  predict $c_{i,j}$, $C_{i,j}$ for all ad campaigns using pretrained click and purchase model;
  set $b_{i,j} = \min((1-\alpha_j)mCPC_j, \text{remaining budget})$;
  compute ranking score: $s_{i,j} = c_{i,j}(b_{i,j} + r * C_{i,j}R_j)$;
  determine the actual CPC for winners;
  update the remaining budget for winners depending on user actions(click);
  if *time is a check point* **then**
    update α_j;
  end
end

Algorithm 1: Bidding and Ranking Algorithm for the simple model without performance constraints

Remark 3.2. *Note that we derived above ranking score using first price revenue in the modeling. However, in practice, one can use any auction mechanism to determine the actual cost per click a winner has to pay. If it is the first price auction, then the winner will pay his bid. If it is the generalized second price auction, then the winner will pay the amount so that it matches the the second ranking score.*

3.5 Extension to incorporate performance constraints

In this section, we discuss briefly how to solve for the model when we include some performance constraints into the model. The idea can be extended naturally if we want to include more platform-specified features into the model. Suppose that we have the goal for total number of clicks, then relaxed LP model is the following:

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{N}\sum_{j=1}^{M} c_{i,j}Z_{i,j} + c_{i,j}C_{i,j}W_{i,j}R_j * r \\
\text{s.t} \quad & \sum_{j=1}^{M} W_{i,j} \le 1 \ \ \forall i \\
& \sum_{i=1}^{N} c_{i,j}Z_{i,j} \le B_j \ \ \forall j \\
& Z_{i,j} \le mCPC_j W_{i,j} \ \ \forall i,j \\
& \sum_{i=1}^{N}\sum_{j=1}^{N} c_{i,j}W_{i,j} \ge G_{\text{click}}N \\
& W_{i,j}, Z_{i,j} \ge 0 \ \ \forall i,j
\end{aligned}
$$

Let $\theta_c \ge 0$ be the dual variable for the click constraints, the dual problem is the following:

$$
\begin{aligned}
& \sum_{i=1}^{N}\beta_i + \sum_{i=1}^{M}\alpha_j - NG_{\text{click}}\theta_c \\
& r_{i,j} + \alpha_j c_{i,j} \ge c_{i,j} \ \ \forall i,j \\
& \beta_i - r_{i,j}mCPC_j - \theta_c c_{i,j} \ge c_{i,j}C_{i,j}R_j * r \ \ \forall i,j \\
& \beta_i, \alpha_j, r_{i,j} \ge 0 \ \ \forall i,j
\end{aligned}
$$

Proposition 3.3. *Suppose that the primal problem is feasible, then a dual optimal solution α_i^*, β_j^*, $r_{i,j}^*$s' and θ_c^* will satisfy the following condition:*

- $\alpha_j^* \in [0,1] \ \ \forall j$
- $r_{i,j}^* = (1-\alpha_j^*)c_{i,j}$
- $\beta_i^* = \max_j c_{i,j}((1-\alpha_j^*)mCPC_j + \theta_c^* + C_{i,j}R_j * r) \ \ \forall i,j$

Thus, the only change in the ranking function is that we have an additional penalty term $\theta_c c_{i,j}$. It is a parameter corresponding to how well the platform achieves the goal for total number of clicks. If $\theta_c = 0$, then it corresponds to the original simple model. If $\theta_c > 0$, then compared with the previous ranking score, the new ranking function rewards those ads with higher CTR. Similar as before, we can have an adaptive control over θ_c: $\theta_c(t+1) = \theta_c(t)\exp(\kappa(\hat{G_{\text{click}}} - G_{\text{click}}))$, where $\hat{G_{\text{click}}}$ is the actual click percentage among all the impressions between time t and $t+1$ and G_{click} is the planned click percentage between time t and $t+1$. The control pace for θ_c could be slower than α_js'. For instance, it can be updated daily or even weekly.

For completeness, we list the corresponding performance constraints and the ranking functions in the following table. The key insight here is that the dual variables play an important role to adaptively control the discrepancy between actual performance and target performance.

category	constraints
click	$\sum_{i,j} c_{i,j}W_{i,j} \ge G_{\text{click}}N$
dual	ranking function
θ_c	$c_{i,j}((1-\alpha_j)mCPC_j + \theta_c + C_{i,j}R_j * r)$
category	constraints
shared purchase	$\sum_{i,j} c_{i,j}W_{i,j}C_{i,j}R_j * r \ge G_p$
dual	ranking function
θ_p	$c_{i,j}((1-\alpha_j)mCPC_j + C_{i,j}R_j * (r+\theta_p))$
category	constraints
ROI	$\sum_{i,j} c_{i,j}W_{i,j}C_{i,j}R_j \ge G_r(\sum_{i,j} c_{i,j}Z_{i,j})$
dual	ranking function
θ_r	$c_{i,j}((1-\alpha_j - G_r\theta_r)mCPC_j + C_{i,j}R_j(\theta_r + r))$

Table 1: Performance constraints and corresponding ranking functions

4 EXPERIMENTS

We used a real world dataset from an e-commerce company and designed some experiments to test the proposed model. This e-commerce company provides opportunities for sellers to bid for ads positions in order to promote their products. In their current production system, it ranks the ads approximately by their expected bidding revenue: click through rate times the bidding. Meanwhile, pacing factor is considered in the ranking function. For privacy reason, the exact tanking function will be revealed here. The company shares a certain percentage of the purchase revenue with the sellers if an ad is clicked and purchased. The raw dataset contains daily visit logs for 2 weeks. We have extracted the following 4 types of information: (1) search session information: time of impression and search query; (2) ads related information: name, description, price, historical click and purchase information,...etc; (3) auction information: budget, predicted CTR, bid, max bid, pacing factor, ...etc; (4) label information: actual click and purchase label. We separated the data into training data (for the purchase model) and auction data and they are disjoint.

It is necessary to train a purchase model before providing predicted CVR values for each ads. For predicted CTR values, they have been logged in the dataset and we reused those values. After that, we simulated the auction on the logged dataset using both current model (implemented in the system) and proposed model. The details are discussed in the following subsections.

4.1 Training CVR model

We modeled purchase rate: $P(\text{purchase}|clicked, ad, query)$ using logistic regression. The features we used for the model are: ads id, timestamp, position, price, title, category, description, query relevance, historical feedback of the ads. We did not use raw texts as sparse features directly. Instead, we used the pretrained word2vec model [2] and extracted the learned vector representation for each word. The original dimension of each word vector is 300. To further reduce the dimension, we applied SVD to the word matrix and used the top 10 dimensional vector for each word. The query relevance features measure the relevancy between the query entry and the ads via their category, title and description.

Logistic regression was implemented in vowpal wabbit (VW) [3]. In this work, we did not focus on obtaining the best predictive models.

[2] http://mccormickml.com/2016/04/12/googles-pretrained-word2vec-model-in-python/

[3] https://github.com/JohnLangford/vowpal_wabbit

4.2 Auction Data

We collected 7 full days' auction data. They were preprocessed so that each search query has exactly 8 ads for impression and this made the testing dataset a downsampled subset. One (common) big challenge to rerun the auction is that losers were never logged. Only the winning ads that have been shown to the users were logged. This implies that we can never fully recover the auction at logged timestamp. Another inherent problem is the counterfactual nature of rerunning the auction. The user actions (click and purchase) can not be observed after we rerank the ads.

To tackle these problems, we assume that the logged click and purchase labels remain the same after the ads have been reranked. This is a reasonable assumption since we limit the ads in a single search result page and the click and purchase rate are continuous with respect to the position. Meanwhile, after we reranked these 8 ads for each search session, bidding revenue and shared purchase revenue were computed only from the first 4 ads slot. In this way, we treated the last 4 advertisers as losers.

Still, this did not resolve the case where the original losers could have won the auction under the new ranking rules and their budget usage throughout the day would be completely different. This is one of the limitations of current offline evaluation. Alternative approach includes simulating the click and purchase label. However, this will introduce high bias unless the the click and purchase predictive model are fairly accurate.

4.3 Evaluation metrics

Bidding revenue and shared purchase revenue are direct measures of the platform, and rescaled shared purchase revenue is a measure for the advertisers. In addition, we will use a few indirect measures: (1) effective CPC (eCPC) (2) shared revenue per purchase (RP) (3) competitive ratio (CR) and (4) ROI defined as follows:

$$eCPC = \frac{\text{total bidding revenue}}{\text{total number of clicks}}$$

$$RP = \frac{\text{total shared purchase revenue}}{\text{total number of purchases}}$$

$$CP = \frac{\text{total bidding revenue}}{\text{total first price revenue}}$$

$$ROI = \frac{\text{total purchase revenue}}{\text{total bidding revenue}}$$

eCPC is to measure how expensive each click is across all the advertisers. High eCPC is good for the platform, but bad for the advertisers. PR is to measure the gain for the platform from each purchase/conversion of an ad across all purchases. CR is to measure the non-linearity of the second price auction. If it is of high value, then the auction is close to first price auction and more bidding revenue can be expected. ROI is to measure the return for advertisers, and this is what they really care about.

4.4 Experiment I: sufficient budget

For the first set of experiments, we used logged budget for advertisers to simulate the auction. In the current system, the advertisers can specify their maximal CPC, and if they do not, we used the historical maximal bid to approximate this quantity. Note that labels for purchase and click are from the logged dataset, the maximal number of clicks each advertiser can have is also fixed no matter how we rerun the auction. In the original dataset, the bidding cost did exceed the budget and this property will remain true in the simulation for most of the advertisers.

The default setting of the parameter is the following: $\forall j, \alpha_j(0) = 0.5$ and the updating frequency is every 20000 search queries. Under this setting, we bid only half of the max bid at the beginning of the day. Under the uniform pacing scheme, the planned budget spending during each time interval is the same: $B_j(t) = \frac{B_j}{T}$. In the actual experiments, we adopted another uniform pacing scheme: $B_j(t+1) = \frac{B_j - S_j(t)}{T-t}$. This is a uniform pacing based on the remaining budget.

4.5 Performance under the default setting

Using both proposed and current method (implemented in the production system), we reran the auction simulation on 7 day's auction data. For each day, we plot the change of percentage for different metrics compared to the current performance. The metrics include the bidding revenue, shared purchase revenue, total revenue and rate of return (ROI) for advertisers.

$$\frac{\text{metric}_{\text{proposed}} - \text{metric}_{\text{current}}}{\text{metric}_{\text{current}}}$$

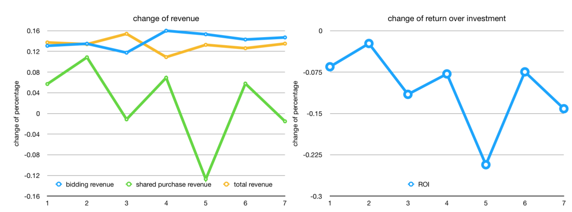

Figure 1: comparison between current model and proposed model

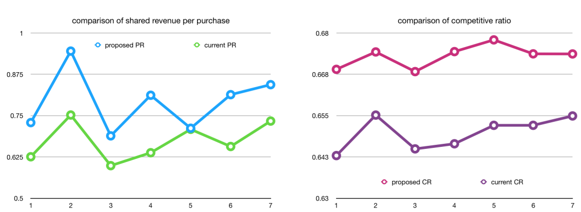

Figure 2: comparison between current model and proposed model

From Figure 1, both bidding revenue and total revenue outperform the current system by at least 10%. The shared purchase revenue also improves most of the time. It was found that competitive ratio increased from 64% (in the current system) to 67% (in the proposed model). This implies that even with the same first price revenue, the proposed model will get more bidding revenue. Moreover, under the proposed model, the shared revenue per purchase (RP) is consistently higher than the current model. Thus, on average, the proposed ranking function is biased towards ads with more valuable conversion. These statistics can be found in Figure 2.

However, the effective cost per click increased (23) compared to the current system (20). Meanwhile, the ROI for the advertisers decreased slightly using the proposed model. This suggests that the default setting greatly improves the performance of the internal platform at a little cost of sacrificing the performance of the advertisers. In the next subsections, we will see how to balance the performance of these two players by tuning parameters of the proposed model.

4.5.1 Effect of click penalty. The click penalty is a dual variable corresponding to the constraint for the total number of clicks. It is designed to increase the total number of clicks for the ad system. In particular, the platform can adjust this click penalty parameter in the ranking function according to the discrepancy between actual total number of clicks and target total number of clicks. In addition to the default setting of parameters, we varied the click parameter from between 0 and 10 and plot the corresponding change of percentage compared to the current system in terms of bidding revenue, shared purchase revenue, total revenue and ROI.

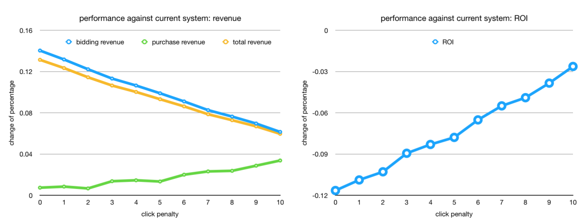

Figure 3: varying click penalty

It can be seen that increasing the click penalty parameter improves the performance of the advertisers. In particular, ROI achieves the same level of current system when α increases to 10. However, it slightly decreases the total bidding revenue in the proposed model. However, as one can see from Figure 3, within range $(0, 10)$, the bidding revenue and the total revenue still outperforms the current model.

4.5.2 Effect of shared purchase percentage in ranking. Shared percentage of purchase revenue is a dual variable to the shared purchase revenue constraint. It is designed to improve the actual purchase revenue for the advertisers. We varied this percentage between 0.01 and 0.45 and plot the change of percentage compared to the current system in terms of bidding revenue, shared purchase revenue total revenue and ROI.

From Figure 4, one sees that both purchase revenue and ROI increase steadily when the shared revenue percentage in the ranking function is increased. However, the bidding revenue and the total revenue becomes worse and even worse than the current system

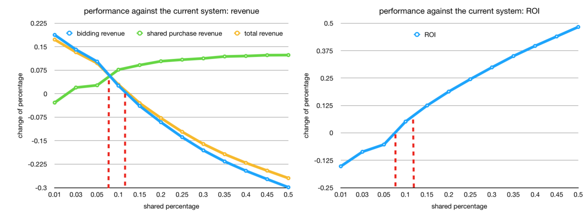

Figure 4: varying shared revenue percentage

when $r > 0.125$. Within range $(0.1, 0.125)$, the proposed model outperforms the current system for both players: the internal platform and advertisers. This is a good region for a global optimization point of view.

4.5.3 Analysis. The above two experiments demonstrate that varying click penalty and shared percentage of purchase revenue in ranking are effective ways to balance the performance of the internal platform and the advertisers. It reiterates the conflicting nature of modeling DSP and SSP. Indeed, one can view the internal platform and the advertisers are natural adversaries, and modeling the system is playing a min max game against advertisers. From the above experimental results, we see that there exists a range of parameter setting for the proposed model such that both performance of the platform and advertisers are better than the current system. Practically, it implies that the proposed model can be further optimized with respect to these model hyper-parameters.

4.6 Experiment II: tight budget

For the second set of experiments, we only used a proportion of actual cost each advertisers incur in the current system as their budget. We want to investigate the performance of the proposed model with a tight budget. As we have mentioned before, when the budget is sufficient, bidding with small α is optimal. However, under tight budget, this could possibly make the advertisers bid higher in the early period of the day and lose the opportunity to exposures due to exhaustion of the budget. We found that there are 24 advertisers in our downsampled auction dataset that have been clicked for at least 10 times under the current model. Their performance will be studied (at a micro level) under different setting of α.

4.6.1 Budget ratio 0.5. We first modified the budget for each advertiser to be $0.5 * \max(\text{spending}, mCPC)$, where spending is the amount that has been spent under the current system, and *mCPC* is the max bid it is willing to pay. The global performance of all advertiser is summarized in Table 2.

It is observed that under the tight budget, small initial α still generates higher bidding revenue from a global perspective (across all the advertisers). We suspect that this is due to the sparse click nature of the dataset, where most advertisers are clicked a few times a day. Thus from the internal platform's perspective, it is still optimal to bid higher with small initial α. The proposed method outperforms both current method and Metha's bidding modification method from the perspective of the platform (the total revenue

Mode	Bid R	Purchase R	Total R	ROI
0.1	1882.88	166.11	2048.99	2.94
0.5	1846.75	177.50	2024.25	3.20
0.8	1811.58	180.56	1992.14	3.32
current	1684.77	168.30	1853.07	3.32
Mehta[13]	1511.49	184.23	1695.72	4.06

Table 2: Global performance with varying initial α

generated by the auction). However, from the perspective of advertisers, Mehta's bidding modification achieves the best return over investment rate. When $\alpha_0 = 0.8$, the rate of return under the proposed model is the same as the current system.

To study the effectiveness of the proposed model, we will study the performance of those 24 advertisers, who have been clicked multiple times throughout a day. In particualr, we tracked their cost per click for each query throughout the day with different choice of parameters of the proposed model. Intuitively, with small initial α, the model will exhaust the budget more quickly and thus their budget utilization curve is concave. With big α, the model will bid lower initially and thus their budget utilization curve is convex. We confirmed this pattern in Figure 5, where we record the global budget usage throughout the day. When $\alpha = 0.5$, the budget utilization curve is almost linear and it corresponds to smooth delivery. Therefore, for those competitive advertisers, it is optimal to use larger initial αs' with guarantee of almost full budget and good pacing under the proposed model.

We also compared the campaign lifetime, number of clicks and the maximal bid overall search queries of these 24 repeatedly clicked advertisers between the current model and the proposed model with $\alpha_0 = 0.5$. We found that using the proposed model, these advertisers on average bid lower and thus have lower cost per click (CPC) for each query and have longer campaign life. Moreover, they have more clicks under the proposed model. The evidence is supported in Figure 6 and Figure 7.

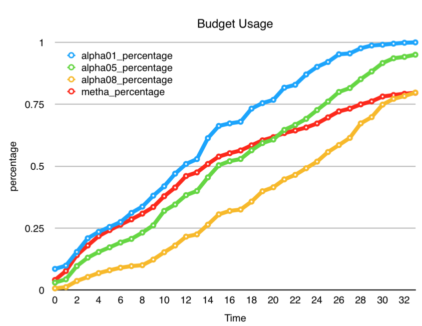

Figure 5: Budget Utilization for multiple clicked advertisers

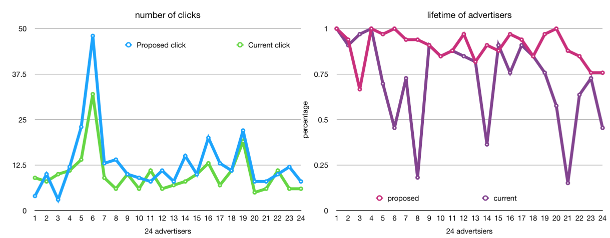

Figure 6: Performance of multiple clicked advertisers

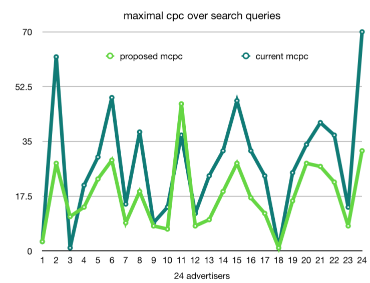

Figure 7: maximal bid over search queries for multiple clicked advertisers

5 FUTURE WORK

A simple revenue maximization model is proposed for an internal ads platform that functions as both DSP and SSP. The model captures the adversarial nature of these two players. Extensive experiments have demonstrated that with proper tuning of model hyper-parameters, performance of both internal platform and advertisers can be improved. There are still a few limitations in this work and we plan to pursue further investigation. From a theoretical level: is there an equilibrium solution for the model? From a practical perspective: (1) if the advertisers have not specified the maximal CPC, how to set this upper bound and further optimize this upper bound? (2) online evaluation (A/B testing) of the proposed model (3) extend to the nonlinear modeling of auction revenue. We have mentioned that using first price revenue in the model gives us an easy and interpretable solution structure. Is it still possible to have an efficient and scalable ads bidding/allocation rule under the varying bidding landscape still remains open.

REFERENCES

[1] Zoe Abrams, Ofer Mendelevitch, and John Tomlin. 2007. Optimal delivery of sponsored search advertisements subject to budget constraints. In *Proceedings of the 8th ACM conference on Electronic commerce*. ACM, 272–278.
[2] Deepak Agarwal, Souvik Ghosh, Kai Wei, and Siyu You. 2014. Budget pacing for targeted online advertisements at linkedin. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 1613–1619.
[3] Olivier Chapelle. 2014. Modeling delayed feedback in display advertising. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 1097–1105.
[4] Ye Chen, Pavel Berkhin, Bo Anderson, and Nikhil R Devanur. 2011. Real-time bidding algorithms for performance-based display ad allocation. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 1307–1315.

[5] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. 2016. Wide & deep learning for recommender systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*. ACM, 7–10.

[6] Ying Cui, Ruofei Zhang, Wei Li, and Jianchang Mao. 2011. Bid landscape forecasting in online ad exchange marketplace. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 265–273.

[7] Brian Dalessandro, Claudia Perlich, Ori Stitelman, and Foster Provost. 2012. Causally motivated attribution for online advertising. In *Proceedings of the Sixth International Workshop on Data Mining for Online Advertising and Internet Economy*. ACM, 7.

[8] Sahin Cem Geyik, Sergey Faleev, Jianqiang Shen, Sean O'Donnell, and Santanu Kolay. 2016. Joint Optimization of Multiple Performance Metrics in Online Video Advertising.. In *KDD*. 471–480.

[9] Paul Grigas, Alfonso Lobos, Zheng Wen, and Kuang-chih Lee. 2017. Profit Maximization for Online Advertising Demand-Side Platforms. *arXiv preprint arXiv:1706.01614* (2017).

[10] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: A Factorization-Machine based Neural Network for CTR Prediction. *arXiv preprint arXiv:1703.04247* (2017).

[11] Kuang-Chih Lee, Ali Jalali, and Ali Dasdan. 2013. Real time bid optimization with smooth budget delivery in online advertising. In *Proceedings of the Seventh International Workshop on Data Mining for Online Advertising*. ACM, 1.

[12] H Brendan McMahan, Gary Holt, David Sculley, Michael Young, Dietmar Ebner, Julian Grady, Lan Nie, Todd Phillips, Eugene Davydov, Daniel Golovin, et al. 2013. Ad click prediction: a view from the trenches. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 1222–1230.

[13] Aranyak Mehta, Amin Saberi, Umesh Vazirani, and Vijay Vazirani. 2007. Adwords and generalized online matching. *Journal of the ACM (JACM)* 54, 5 (2007), 22.

[14] Michael Ostrovsky and Michael Schwarz. 2011. Reserve prices in internet advertising auctions: A field experiment. In *Proceedings of the 12th ACM conference on Electronic commerce*. ACM, 59–60.

[15] Jian Xu, Kuang-chih Lee, Wentong Li, Hang Qi, and Quan Lu. 2015. Smart pacing for effective online ad campaign optimization. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2217–2226.

[16] Yong Yuan, Feiyue Wang, Juanjuan Li, and Rui Qin. 2014. A survey on real time bidding advertising. In *Service Operations and Logistics, and Informatics (SOLI), 2014 IEEE International Conference on*. IEEE, 418–423.

[17] Weinan Zhang, Shuai Yuan, and Jun Wang. 2014. Optimal real-time bidding for display advertising. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 1077–1086.