User Response Learning for Directly Optimizing Campaign Performance in Display Advertising

Kan Ren†, Weinan Zhang†,*, Yifei Rong♯, Haifeng Zhang♮, Yong Yu†, Jun Wang‡
†Apex Data and Knowledge Management Lab, Shanghai Jiao Tong University
‡University College London, ♯YOYI Inc., ♮Peking University
{kren,wnzhang}@apex.sjtu.edu.cn, yifei.rong@yoyi.com.cn

ABSTRACT

Learning and predicting user responses, such as clicks and conversions, are crucial for many Internet-based businesses including web search, e-commerce, and online advertising. Typically, a user response model is established by optimizing the prediction accuracy, e.g., minimizing the error between the prediction and the ground truth user response. However, in many practical cases, predicting user responses is only part of a rather larger predictive or optimization task, where on one hand, the accuracy of a user response prediction determines the final (expected) utility to be optimized, but on the other hand, its learning may also be influenced from the follow-up stochastic process. It is, thus, of great interest to optimize the entire process as a whole rather than treat them independently or sequentially. In this paper, we take real-time display advertising as an example, where the predicted user's ad click-through rate (CTR) is employed to calculate a bid for an ad impression in the second price auction. We reformulate a common logistic regression CTR model by putting it back into its subsequent bidding context: rather than minimizing the prediction error, the model parameters are learned directly by optimizing campaign profit. The gradient update resulted from our formulations naturally fine-tunes the cases where the market competition is high, leading to a more cost-effective bidding. Our experiments demonstrate that, while maintaining comparable CTR prediction accuracy, our proposed user response learning leads to campaign profit gains as much as 78.2% for offline test and 25.5% for online A/B test over strong baselines.

1. INTRODUCTION

Real-time bidding (RTB) based display advertising has gained significant popularity since its emergence in 2009 [20]. As a key advantage over previous contextual advertising paradigm, RTB enables impression-level ad inventory evaluation and user targetting, which largely improves the efficiency of resource reallocation between the buy side (advertising budget) and the sell side (publishers ad inventory).

In RTB display advertising, each time when a user visits a publisher's site (a webpage or a mobile App page), a bid request for the corresponding ad display opportunity, along with its information about the underlying user, the domain context, and the auction, is sent to each advertiser for bid via an ad exchange. In order to calculate the bid, the advertiser first predicts the user response of that ad display, i.e., how likely the user is going to click or convert, which is normally measured by the predicted probability of click (CTR) or conversion (CVR). The advertiers should bid higher and allocate more budget on the ad inventory that has higher CTR or CVR [22].

Typically, the bid optimization is done in a sequential basis. First, the CTR/CVR[1] estimation is formulated as a binary regression problem, which can be solved by such as logistic regression with SGD [23] or FTRL learning [18], Bayesian probit regression [9], gradient boosting regression trees [10], and Factorization Machines [21]. The common objective in this stage is to make the estimation as accurate as possible by, for instance, minimizing the error between the predictions and ground truth user responses. Second, once we have obtained the user response prediction, we will use it as an input for optimizing the bid based on other considerations including campaign budget, market price, etc [14, 32, 31, 22].

However, such sequential optimization is not ideal. According to the Bayesian decision theory [3], the learning of the user response model should be informed by the final bidding utility. For instance, the required accuracy of the CTR prediction would not be the same throughout the range of the prediction $[0, 1]$ as there is a cost (negative utility) for the advertiser to win an impression if no click, but no cost (zero utility) for losing one. The value of clicks also varies across campaigns; and it would be good if the CTR learning can tailor its efforts more towards those higher-valued cases and make them better predicted. More importantly, the user response prediction is correlated with the second price auction in RTB — if won an auction, the advertiser pays the market price [2], i.e., the highest bid from competitors, and then obtain the payoff from the user conversions driven by the ad. Therefore, the market price and the competition have a significant impact on the campaign performance. If the performed bid is in a highly competitive situation, it is of low confidence to predict whether the advertiser will win the ad auction or not; thus the optimization of the CTR prediction in such case should be more focused and fine-tuned than that in the less competitive case.

In this paper, we reformulate the CTR learning problem in the context of the second price auction. Instead of independently training the CTR estimator, we consider the CTR estimation as a part of bid optimization and its prediction directly determines the auction results and thus the

*W. Zhang is the corresponding author.

[1] In this paper, we focus on the CTR estimation, while the CVR estimation can be done by following the same token.

CIKM'16 , October 24-28, 2016, Indianapolis, IN, USA
© 2016 ACM. ISBN 978-1-4503-4073-1/16/10...$15.00
DOI: http://dx.doi.org/10.1145/2983323.2983347

advertising performance and vice versa. With this in mind, we learn a CTR model by directly optimizing the campaign performance, where the settings with or without budget constraints are studied. The resulting gradient functions for learning a CTR model reveal that learning weights should reflect the market competition of the auction. Specifically, the higher probabilistic density the market price is at the bid price, the higher learning weight is set on the data instance. This naturally provides a more learning focus on the instances where the competition is high. The fine tuned model would be able to bid close to the market price, resulting in more cost-effective bidding. Note that in the second price auction, the game theoretical approach and the rational assumption lead to a truthful bidding, independent of market price [11]. However, empirical studies have shown that advertisers are quite often not rational in RTB [29] and in this paper, we take a stochastic approach by following [2].

We have conducted large-scale offline experiments based on real-world data with various budget and bidding strategy settings. We find that our solutions have as much as 78.2% profit gain over baselines. Also we have deployed our CTR predictor training schemes onto a commercial RTB platform. The online A/B testing results demonstrate that our proposed CTR predictor learning directly leads to 25.5% campaign profit gain over a widely used CTR model.

The rest of this paper is organized as follows. In Section 2 we compare related work with ours. In Section 3, we propose the CTR training framework optimizing campaign performance. Our experiment is given in Section 4. We conclude the paper with the future work in Section 5.

2. RELATED WORK

User Response Prediction. The user response prediction, such as the click-through rate (CTR) estimation or the conversion rate (CVR) estimation, has become a core research problem in real-time display advertising [14, 18, 26]. The response prediction is a probability estimation task [19] which models the interest of users towards the content of publishers or the ads, and is used to derive the budget allocation of the advertisers [23]. Typically, the response prediction problem is formulated as a regression problem with prediction likelihood as the training objective [23, 9, 1, 21]. From the methodology view, linear models such as logistic regression [14] and non-linear models such as tree-based model [10] and factorization machines [19, 21] are commonly used. Other variants include Bayesian probit regression [9], FTRFL [24] in factorization machine, and convolutional neural network learning framework [17]. Normally, area under ROC curve (AUC) and relative information gain (RIG) are common evaluation metrics for CTR prediction accuracy [9]. Recently, the authors in [4, 25] pointed out that such metrics were not good enough for evaluating CTR predictor in RTB based advertising because of the subsequent bidding and auctions. In this paper, we use a logistical regression as a working example and go one step further over [4] to reformulate the CTR estimation learning by directly optimizing campaign performance (profit).

Bidding Strategy. With the estimated CTR/CVR, the advertisers would be able to assess the value of the impression and perform a bid. The auction theory [7] proves that truthful bidding, i.e., bidding the action value times the action rate, is the optimal strategy in the second price auction [14]. However, with budget and auction volume constraints, the truthful bidding may not be optimal [30]. The linear bidding strategy [22] is widely used in industry, where the bid price is calculated via the predicted CTR/CVR multiplied by a constant parameter tuned according to the campaign budget and performance. The authors in [5] proposed a bidding function with truthful bidding value minus a tuned parameter. A lift-based bidding strategy was recently proposed in [28] where the bid price was determined by the user CVR lift after seeing the displayed ad.

However, the impact of market price distribution, i.e. bid landscape, is not studied in above work, and the final utility of the campaign is not considered in the optimization objective, which may result in some unfavorable statistics such as relatively high eCPC and low return-on-investment ratio (ROI). The authors in [15] combined the winning rate estimation and the winning price prediction together and deployed the estimation results in different bidding strategies towards different business demands. The authors in [13] embedded a budget smoothing component into a bid optimization framework. In [32, 31], using a CTR estimation as an input, the authors proposed non-linear bidding functions. Our work is different from the above work as we directly model CTR learning as part of bid optimization for campaign profit maximization.

Bid Landscape. Bid landscape forecasting refers to predicting the distribution of market price for a type of ad inventory [6]. The advertisers use it to calculate the winning rate given a bid and decide the final bid price. Several winning function forms were hypothesized in [15, 32] to directly induce the optimal bidding functions. A campaign-level forecasting system with tree models was presented in [6]. The authors in [12] conducted an error handling methodology to improve the efficiency and reliability of the bid landscape forecasting system. As advertisers only know the statistics (market price, user clicks etc.) from their winning impressions, the authors in [27] proposed a solution to handle such data censorship in market price prediction. Later we will the show market price distribution indeed plays an essential role in CTR model learning for campaign profit optimization, which has never been proposed.

To sum up, all the existing learning frameworks in RTB consider the user response prediction and bidding optimization as two separated parts, while in our paper, we model them as a whole and perform a novel joint optimization.

3. THE PROPOSED SOLUTION

In this section, we formulate the CTR model learning to directly optimize profit for a performance campaign (the goal is to acquire new customers in order to generate sales).

3.1 Problem Setup and Objective Function

We consider the following performance campaign setting: advertisers typically employ a DSP (demand side platform), which connects to ad exchanges, to deliver their ads. For a given ad campaign, when a bid request from an ad exchange hits pre-specified target rules (targeted domains or audiences), the DSP predicts its CTR/CVR and then calculates a bid in real-time using the predicted CTR/CVR. Without loss of generality, we take clicks as the expected user actions and the problem is to learn the user response (CTR) model by considering the entire bidding process and the final utility of the bidding.

A bid request contains various information of an ad display opportunity, including the information of the underlying user, location, time, user terminal, browser, the contextual information about the webpage etc. Together with the features extracted from the campaign itself, we construct the high-dimensional feature vector for the bid request, denoted as $\boldsymbol{x}$. We also use $p_{\boldsymbol{x}}(\boldsymbol{x})$ to denote the probability

Table 1: Notations and descriptions

Notation	Description
v	The pre-defined value of positive user response.
y	The true label of user response.
$\boldsymbol{x}$	The bid request represented by its features.
$p_{\boldsymbol{x}}(\boldsymbol{x})$	The probability density function of $\boldsymbol{x}$.
z	The market price.
$p_z(z)$	The probability density function of z.
$\boldsymbol{\theta}$	The weight of CTR estimation function.
$f_{\boldsymbol{\theta}}(\boldsymbol{x})$	the CTR estimation function to learn.
$b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))$	The bid price determined by the estimated CTR, b for short.
$R_{\boldsymbol{\theta}}(\cdot)$	The utility function.
$w(b)$	The winning probability given bid price b.
$c(b)$	The expected cost given bid price b if winning.

distribution of the input feature vector $\boldsymbol{x}$ that matches the campaign target rules .

Formally, the CTR estimation is denoted as a function $p(y=1|\boldsymbol{x}) \equiv f_{\boldsymbol{\theta}}(\boldsymbol{x})$ mapping from feature $\boldsymbol{x}$ to the probability of a click, where $y \in \{0,1\}$ is a binary variable indicating whether a user click occurs (1) or not (0). The function is parameterized by $\boldsymbol{\theta}$. We also denote the true value of a click as v, which is pre-specified by the advertiser for a given campaign[2].

Next we define the context where the CTR estimator is situated. Previous studies have specified the bidding strategy as a function $b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))$ mapping from the predicted CTR (or other KPIs) $f_{\boldsymbol{\theta}}(\boldsymbol{x})$ to the bid price [14, 22, 32]. Essentially, the mapping follows a sequential dependency assumption $\boldsymbol{x} \rightarrow f_{\boldsymbol{\theta}}(\boldsymbol{x}) \rightarrow b$ proposed by [32, 31]. In this paper, we follow the same formulation. For simplicity, we use $b(\cdot)$ to represent the bidding function, but also occasionally use b to directly represent the bid price.

Once the DSP sends out the bid b, the ad exchange hosts a second-price auction [11] and decides who is going to win the auction. The probability of winning an auction is influenced by the bid price b and the stochastic market price z with an underlying p.d.f. $p_z(z)$; we use $w(b)$ to denote the probability of winning as:

$$w(b) = \int_0^b p_z(z)dz, \tag{1}$$

which is the probability that the bid b is higher than the market price z [11]. Later we will discuss the important role of $p_z(z)$ in our CTR learning.

If the bid wins the auction, the advertiser pays the cost, which is the market price z. We denote the expected cost in the second price auction as

$$c(b) = \frac{\int_0^b z p_z(z)dz}{\int_0^b p_z(z)dz}, \tag{2}$$

which is essentially the expected market price when winning the auction [11]. Once we have defined the bidding function b, the value of a click v, and the winning rate w, the expected cost c, we are ready to define a general form of the utility function as $R_{\boldsymbol{\theta}}(\boldsymbol{x}, y; b, v, c, w)$ for a given $(\boldsymbol{x}, y)$ 2-tuple in the training data (all the received historical impressions).

Our task of building a user response model can be thus formulated as to learn the optimal parameter $\boldsymbol{\theta}^*$ so that the expected utility will be maximized:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \int_{\boldsymbol{x}} R_{\boldsymbol{\theta}}(\boldsymbol{x}, y; b, v, c, w) p_{\boldsymbol{x}}(\boldsymbol{x}) d\boldsymbol{x}, \tag{3}$$

[2]Can be calculated as the probability of a conversion from the click multiplied by the value of the converted sale.

where we do not include a campaign budget constraint for optimizing the CTR model as we will show later in Sections 3.2 and 3.3 that the utility function $R_{\boldsymbol{\theta}}(\cdot)$ has included the cost, while the true value of a click v limits the max bid. The budget constraint is, however, incorporated in bid optimization [32], which will be discussed in Section 3.6. For readability, our notations are summarized in Table 1.

3.2 Gradient for Expected Utility

To solve Eq. (3), utility function $R_{\boldsymbol{\theta}}(\cdot)$ can be naturally defined as the expected direct profit from the campaign:

$$R_{\boldsymbol{\theta}}^{\mathrm{EU}}(\boldsymbol{x}, y) = [vy - c(b(f_{\boldsymbol{\theta}}(\boldsymbol{x})))] \cdot w(b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))), \tag{4}$$

where to simplify our notation, we drop the dependency of b, v, c, w for $R_{\boldsymbol{\theta}}^{\mathrm{EU}}(\boldsymbol{x}, y)$. The expectation is with respect to whether winning or not, where no winning has zero utility. Recall that, in the training set defined as D, each sample is represented as a 2-tuple as $(\boldsymbol{x}, y)$, where $\boldsymbol{x}$ denotes the feature vector of the bid request, and y denotes the indicator whether user action (click) occurs. The overall expected direct profit [4] of all the auctions can be calculated by replacing Eqs. (1) and (2) into Eq. (4) as

$$\begin{aligned}
&\sum_{(\boldsymbol{x},y)\in D} R_{\boldsymbol{\theta}}^{\mathrm{EU}}(\boldsymbol{x}, y) \\
&= \sum_{(\boldsymbol{x},y)\in D} \left[vy - \frac{\int_0^{b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))} z \cdot p_z(z)dz}{\int_0^{b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))} p_z(z)dz} \right] \cdot \int_0^{b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))} p_z(z)dz \\
&= \sum_{(\boldsymbol{x},y)\in D} \int_0^{b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))} (vy - z) \cdot p_z(z)dz.
\end{aligned} \tag{5}$$

Taking Eq. (5) into Eq. (3) with a regularization term turns our learning problem into convex optimization:

$$\begin{aligned}
\boldsymbol{\theta}^{\mathrm{EU}} &= \arg\min_{\boldsymbol{\theta}} \; - \sum_{(\boldsymbol{x},y)\in D} R_{\boldsymbol{\theta}}^{\mathrm{EU}}(\boldsymbol{x}, y) + \frac{\lambda}{2}\|\boldsymbol{\theta}\|_2^2 \\
&= \arg\min_{\boldsymbol{\theta}} \sum_{\boldsymbol{x}} \int_0^{b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))} (z - vy) \cdot p_z(z)dz + \frac{\lambda}{2}\boldsymbol{\theta}^T\boldsymbol{\theta},
\end{aligned} \tag{6}$$

where the optimal value of $\boldsymbol{\theta}$ is obtained by taking a gradient descent algorithm. The gradient of $R_{\boldsymbol{\theta}}^{\mathrm{EU}}(\boldsymbol{x}, y)$ with regard to $\boldsymbol{\theta}$ is calculated as

$$\begin{aligned}
\frac{\partial R_{\boldsymbol{\theta}}^{\mathrm{EU}}(\boldsymbol{x}, y)}{\partial \boldsymbol{\theta}} = &\overbrace{(b(f_{\boldsymbol{\theta}}(\boldsymbol{x})) - vy)}^{\text{bid error}} \cdot \overbrace{p_z(b(f_{\boldsymbol{\theta}}(\boldsymbol{x})))}^{\text{market sensitivity}} \cdot \\
&\frac{\partial b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))}{\partial f_{\boldsymbol{\theta}}(\boldsymbol{x})} \frac{\partial f_{\boldsymbol{\theta}}(\boldsymbol{x})}{\partial \boldsymbol{\theta}} + \lambda\boldsymbol{\theta},
\end{aligned} \tag{7}$$

and we update for each data instance as $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \frac{\partial R_{\boldsymbol{\theta}}^{\mathrm{EU}}(\boldsymbol{x}, y)}{\partial \boldsymbol{\theta}}$ by above chain rule.

Discussion. Eq. (7) provides a novel gradient update, taking into account both the utility and the cost of a bidding decision (the *bid error* term) as well as the impact from the market price distribution (the *market sensitivity* term). They act as two additional re-weighting functions influencing a conventional gradient update, which is formulated by the remaining terms in the equation. We illustrate their impact in Figure 1. The left subfigure shows the weight from *bid error* against bid price with different user responses ($y=1$ or $y=0$). We see that the update of the CTR model aims to correct a bid towards the true value vy from a training instance, i.e., an optimal model (parameter) would generate a bid close to v for a positive instance, while close to zero for a negative instance. The

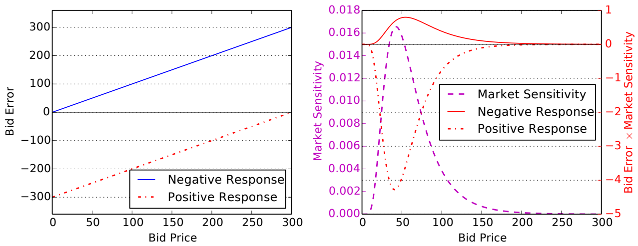

Figure 1: The illustration of the impact from the bid and market price of Expected Utility (EU); click value $v = 300$.

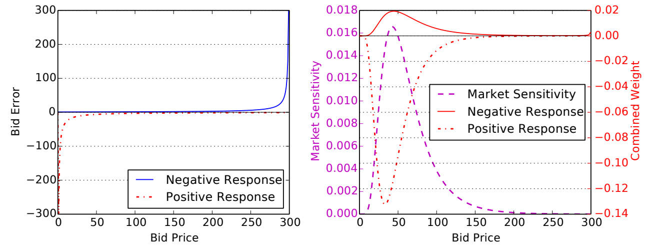

Figure 2: The illustration of the impact from the bid and market price of Risk Return (RR); click value $v = 300$.

right subfigure plots the weight adjustment from the *market sensitivity* term (y-axis left) and the combined weight *bid error* $\times$ *market sensitivity* (y-axis right). We observe that the *market sensitivity* term re-weights the bid error by checking the fitness to the market price distribution; this makes the gradient focused more on fixing the errors (if any) when the bid is close to the market price. This is intuitively correct because when the bid is close to the market price, the competition is high and a small error (win a case that is no click and vice versa) would make a huge difference in terms of the cost and reward. Specifically, for the negative response ($y = 0$), the combined weight $bp_z(b)$ stays positive in order to constantly lower the bid via CTR learning, but its peak location is slightly higher than the mode of market price. For the positive response ($y = 1$), the combining weight $(b - v)p_z(b)$ is negative to push the bid higher to v. Note that the bid is restricted in $[0, v]$ as bidding higher than v is of no advantage than bidding v when optimizing profit.

3.3 Gradient for Risk-Return

Besides the expected utility (EU), we also propose a risk-return (RR) model to balance the risk and return of a bid decision as below:

$$R^{\mathrm{RR}}_{\boldsymbol{\theta}}(\boldsymbol{x}, y) = \Big(\underbrace{\frac{vy}{z}}_{\text{return}} - \underbrace{\frac{v(1-y)}{v-z}}_{\text{risk}} \Big) \cdot w(b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))), \quad (8)$$

where we define that when $y = 1$, the winning utility is $\frac{v}{z}$, which is the ratio between the return and the cost of this transaction; when $y = 0$, the winning utility becomes the penalty for taking risk $\frac{-v}{v-z}$, which is defined as the ratio between the lost $(-v)$ and the gain if winning $(v - z)$. Note that v is always higher than z as $v \geq b > z$. The penalty is very high when bidding for a very low margin (low $v - z$) case. Thus the new optimization objective function is

$$\begin{aligned} \boldsymbol{\theta}^{\mathrm{RR}} &= \arg\min_{\boldsymbol{\theta}} \; - \sum_{(\boldsymbol{x},y)\in D} R^{\mathrm{RR}}_{\boldsymbol{\theta}}(\boldsymbol{x}, y) + \frac{\lambda}{2}\|\boldsymbol{\theta}\|_2^2 \\ &= \arg\min_{\boldsymbol{\theta}} \; - \sum_{(\boldsymbol{x},y)\in D} \int_0^{b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))} \Big(\frac{vy}{z} - \frac{v(1-y)}{v-z} \Big) \cdot p_z(z) dz \\ &\quad + \frac{\lambda}{2}\boldsymbol{\theta}^T\boldsymbol{\theta}, \end{aligned} \quad (9)$$

which leads to the gradient of $R^{\mathrm{RR}}_{\boldsymbol{\theta}}(\boldsymbol{x}, y)$ w.r.t. $\boldsymbol{\theta}$ as

$$\begin{aligned} \frac{\partial R^{\mathrm{RR}}_{\boldsymbol{\theta}}(\boldsymbol{x}, y)}{\partial \boldsymbol{\theta}} = & \Big(\overbrace{-\frac{vy}{b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))} + \frac{v(1-y)}{v - b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))}}^{\text{bid error}} \Big) \cdot \overbrace{p_z(b(f_{\boldsymbol{\theta}}(\boldsymbol{x})))}^{\text{market sensitivity}} \\ & \cdot \frac{\partial b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))}{\partial f_{\boldsymbol{\theta}}(\boldsymbol{x})} \frac{\partial f_{\boldsymbol{\theta}}(\boldsymbol{x})}{\partial \boldsymbol{\theta}} + \lambda \boldsymbol{\theta}. \end{aligned} \quad (10)$$

Discussion. To understand the above gradient, we plot the *bid error*, *market sensitivity* and their combined weight in Figure 2. The RR model is different from the previous EU model in that, the *bid error* turns to *return* when the response is positive, and becomes *risk* when meets a negative response. If $y = 0$ and bid price is high, or if $y = 1$ and bid price is low, the bid error is quite significant to avoid the happening of such cases.

As is shown both in Eqs. (7) and (10), the market price distribution plays an important role in the optimization: with the determined bidding function and CTR estimation function, the gradient is weighted by the probability dense function of market price, which is denoted as $p_z(z)$.

Various bid landscape models can be utilized to model $p_z(z)$, such as the parametric log-normal distribution [6] and Gamma distribution [4]. In this paper, while our model is flexible with various landscape models, we adopt a non-parametric $p_z(z)$ directly obtained from each campaign's winning price data [2].

3.4 Model Realization

Solving the proposed learning objectives (6) and (9) relies on the realization of the bidding function $b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))$, the market price distribution $p_z(z)$ and the CTR estimation function itself $f_{\boldsymbol{\theta}}(\boldsymbol{x})$. In this section, we will discuss the solutions from the proposed two training objectives given some specific implementations of $b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))$, $p_z(z)$ and $f_{\boldsymbol{\theta}}(\boldsymbol{x})$.

Without loss of generality, for the CTR estimation model, we adopt the widely used logistic regression for $f_{\boldsymbol{\theta}}(\boldsymbol{x})$

$$f_{\boldsymbol{\theta}}(\boldsymbol{x}) \equiv \sigma(\boldsymbol{\theta}^T\boldsymbol{x}) = \frac{1}{1 + e^{-\boldsymbol{\theta}^T\boldsymbol{x}}} , \quad (11)$$

and get $\frac{\partial f_{\boldsymbol{\theta}}(\boldsymbol{x})}{\partial \boldsymbol{\theta}} = \sigma(\boldsymbol{\theta}^T\boldsymbol{x})(1 - \sigma(\boldsymbol{\theta}^T\boldsymbol{x}))\boldsymbol{x}$.

For the bidding strategy, we employ a widely used linear bidding function w.r.t. the predicted CTR [22] with a scaling parameter ϕ

$$b(f_{\boldsymbol{\theta}}(\boldsymbol{x})) \equiv \phi \cdot v \cdot f_{\boldsymbol{\theta}}(\boldsymbol{x}). \quad (12)$$

Taking Eqs. (11) and (12) into (7) and (10), respectively, we derive our final gradient of the proposed EU utility:

$$\begin{aligned} \frac{\partial R^{\mathrm{EU}}_{\boldsymbol{\theta}}(\boldsymbol{x}, y)}{\partial \boldsymbol{\theta}} =& \phi v^2(\phi\sigma(\boldsymbol{\theta}^T\boldsymbol{x}) - y) \cdot p_z(b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))) \cdot \\ & \sigma(\boldsymbol{\theta}^T\boldsymbol{x})(1 - \sigma(\boldsymbol{\theta}^T\boldsymbol{x}))\boldsymbol{x} + \lambda\boldsymbol{\theta}, \end{aligned} \quad (13)$$

and that of the RR utility:

$$\begin{aligned} \frac{\partial R^{\mathrm{RR}}_{\boldsymbol{\theta}}(\boldsymbol{x}, y)}{\partial \boldsymbol{\theta}} =& \phi v \Big(- \frac{y}{\phi\sigma(\boldsymbol{\theta}^T\boldsymbol{x})} + \frac{1-y}{1 - \phi\sigma(\boldsymbol{\theta}^T\boldsymbol{x})} \Big) \cdot \\ & p_z(b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))) \cdot \sigma(\boldsymbol{\theta}^T\boldsymbol{x})(1 - \sigma(\boldsymbol{\theta}^T\boldsymbol{x}))\boldsymbol{x} + \lambda\boldsymbol{\theta}, \end{aligned} \quad (14)$$

where the bidding function parameter ϕ acts as a calibration term in bid correction.

3.5 Links to Previous Work

It is of great interest to compare our profit-optimized solutions with the existing ones that optimize the fitness

Table 2: The comparison of the model gradients (without regularization). LR: logistic regreesion, TB: truthful bidding, LB: linear bidding, UM: uniform market price distribution. LR and LR+TB+UM are equivalent (LR+TB reduces to the baseline LR when assuming the uniform market price distribution).

Model Setting	EU (SE) Gradient	RR (CE) Gradient
LR (baseline)	$\frac{\partial \mathcal{L}^{\text{SE}}_{\boldsymbol{\theta}}(\boldsymbol{x},y)}{\partial \boldsymbol{\theta}} = (\sigma(\boldsymbol{\theta}^T\boldsymbol{x}) - y)\cdot\sigma(\boldsymbol{\theta}^T\boldsymbol{x})(1-\sigma(\boldsymbol{\theta}^T\boldsymbol{x}))\boldsymbol{x}$	$\frac{\partial \mathcal{L}^{\text{CE}}_{\boldsymbol{\theta}}(\boldsymbol{x},y)}{\partial \boldsymbol{\theta}} = (\sigma(\boldsymbol{\theta}^T\boldsymbol{x}) - y)\boldsymbol{x}$
LR+TB	$\frac{\partial R^{\text{EU}}_{\boldsymbol{\theta}}(\boldsymbol{x},y)}{\partial \boldsymbol{\theta}} = v^2(\sigma(\boldsymbol{\theta}^T\boldsymbol{x}) - y)\cdot p_z(b(f_{\boldsymbol{\theta}}(\boldsymbol{x})))\cdot\sigma(\boldsymbol{\theta}^T\boldsymbol{x})(1-\sigma(\boldsymbol{\theta}^T\boldsymbol{x}))\boldsymbol{x}$	$\frac{\partial R^{\text{RR}}_{\boldsymbol{\theta}}(\boldsymbol{x},y)}{\partial \boldsymbol{\theta}} = v(\sigma(\boldsymbol{\theta}^T\boldsymbol{x}) - y)\cdot p_z(b(f_{\boldsymbol{\theta}}(\boldsymbol{x})))\cdot\boldsymbol{x}$
LR+TB+UM	$\frac{\partial R^{\text{EU}}_{\boldsymbol{\theta}}(\boldsymbol{x},y)}{\partial \boldsymbol{\theta}} = v^2 l(\sigma(\boldsymbol{\theta}^T\boldsymbol{x}) - y)\cdot\sigma(\boldsymbol{\theta}^T\boldsymbol{x})(1-\sigma(\boldsymbol{\theta}^T\boldsymbol{x}))\boldsymbol{x}$	$\frac{\partial R^{\text{RR}}_{\boldsymbol{\theta}}(\boldsymbol{x},y)}{\partial \boldsymbol{\theta}} = vl(\sigma(\boldsymbol{\theta}^T\boldsymbol{x}) - y)\boldsymbol{x}$
LR+LB	$\frac{\partial R^{\text{EU}}_{\boldsymbol{\theta}}(\boldsymbol{x},y)}{\partial \boldsymbol{\theta}} = \phi v^2(\phi\sigma(\boldsymbol{\theta}^T\boldsymbol{x}) - y)\cdot p_z(b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))) \cdot\sigma(\boldsymbol{\theta}^T\boldsymbol{x})(1-\sigma(\boldsymbol{\theta}^T\boldsymbol{x}))\boldsymbol{x}$	$\frac{\partial R^{\text{RR}}_{\boldsymbol{\theta}}(\boldsymbol{x},y)}{\partial \boldsymbol{\theta}} = \phi v\Big(-\frac{y}{\phi\sigma(\boldsymbol{\theta}^T\boldsymbol{x})} + \frac{1-y}{1-\phi\sigma(\boldsymbol{\theta}^T\boldsymbol{x})}\Big)\cdot p_z(b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))) \cdot\sigma(\boldsymbol{\theta}^T\boldsymbol{x})(1-\sigma(\boldsymbol{\theta}^T\boldsymbol{x}))\boldsymbol{x}$

of the user response data. A logistic regression could be trained with squared error loss to fit user response data:

$$\mathcal{L}^{\text{SE}}_{\boldsymbol{\theta}}(\boldsymbol{x},y) = \frac{1}{2}(y - \sigma(\boldsymbol{\theta}^T\boldsymbol{x}))^2,$$
$$\frac{\partial \mathcal{L}^{\text{SE}}_{\boldsymbol{\theta}}(\boldsymbol{x},y)}{\partial \boldsymbol{\theta}} = (\sigma(\boldsymbol{\theta}^T\boldsymbol{x}) - y)\sigma(\boldsymbol{\theta}^T\boldsymbol{x})(1-\sigma(\boldsymbol{\theta}^T\boldsymbol{x}))\boldsymbol{x}. \quad (15)$$

More commonly, in a binary output case, a logistic regression can be also trained with cross entropy loss:

$$\mathcal{L}^{\text{CE}}_{\boldsymbol{\theta}}(\boldsymbol{x},y) = -y\log\sigma(\boldsymbol{\theta}^T\boldsymbol{x}) - (1-y)\log(1-\sigma(\boldsymbol{\theta}^T\boldsymbol{x})),$$
$$\frac{\partial \mathcal{L}^{\text{CE}}_{\boldsymbol{\theta}}(\boldsymbol{x},y)}{\partial \boldsymbol{\theta}} = (\sigma(\boldsymbol{\theta}^T\boldsymbol{x}) - y)\boldsymbol{x}. \quad (16)$$

We see that our solutions in Eq. (13) and Eq. (14) extend the original gradients in Eq. (15) and Eq. (16) by (i) replacing the user response errors with the bid errors, and (ii) adding the consideration from the market price and the competition. Let us discuss them next.

User Response Errors v.s. Bid Errors. Directly optimizing bid errors is particularly useful in RTB as advertisers might not know exactly the true value of a click. Our solution with a linear bidding function naturally calibrates any discrepancy and adjusts that in the CTR estimation accordingly. To see this, suppose advertisers know exactly the true value v and perform truthful bidding [14, 22]:

$$b(f_{\boldsymbol{\theta}}(\boldsymbol{x})) = v\cdot f_{\boldsymbol{\theta}}(\boldsymbol{x}). \quad (17)$$

With the truthful bidding and taking the logistic regression CTR estimator again, the gradients of the EU (7) and RR (10) utilities are simplified as:

$$\frac{\partial R^{\text{EU}}_{\boldsymbol{\theta}}(\boldsymbol{x},y)}{\partial \boldsymbol{\theta}} = v^2(\sigma(\boldsymbol{\theta}^T\boldsymbol{x}) - y)\cdot p_z(b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))) \quad (18)$$
$$\cdot\,\sigma(\boldsymbol{\theta}^T\boldsymbol{x})(1-\sigma(\boldsymbol{\theta}^T\boldsymbol{x}))\boldsymbol{x} + \lambda\boldsymbol{\theta},$$
$$\frac{\partial R^{\text{RR}}_{\boldsymbol{\theta}}(\boldsymbol{x},y)}{\partial \boldsymbol{\theta}} = v(\sigma(\boldsymbol{\theta}^T\boldsymbol{x}) - y)p_z(b(f_{\boldsymbol{\theta}}(\boldsymbol{x})))\boldsymbol{x} + \lambda\boldsymbol{\theta}, \quad (19)$$

respectively, where the error term in the EU model becomes the same as that of the squared error loss (15), while that of the RR model goes back to the result from the cross entropy loss (16), both of which are weighted by market sensitivity $p_z(b(f_{\boldsymbol{\theta}}(\boldsymbol{x})))$.

Market Price. Note that, we adopt a non-parametric $p_z(z)$ directly obtained from each campaign's winning price data. Furthermore, we discuss in a special case when we have no prior knowledge about the market and assume the market price distribution is uniform:

$$p_z(z) = l, \quad (20)$$

Eq. (18) is simplified as

$$\frac{\partial R^{\text{EU}}_{\boldsymbol{\theta}}(\boldsymbol{x},y)}{\partial \boldsymbol{\theta}} = v^2 l(\sigma(\boldsymbol{\theta}^T\boldsymbol{x}) - y)\cdot\sigma(\boldsymbol{\theta}^T\boldsymbol{x})(1-\sigma(\boldsymbol{\theta}^T\boldsymbol{x}))\boldsymbol{x} + \lambda\boldsymbol{\theta} \quad (21)$$

which becomes equivalent to the traditional LR learning with squared loss as in Eq. (15). Eq. (19) is simplified as:

$$\frac{\partial R^{\text{RR}}_{\boldsymbol{\theta}}(\boldsymbol{x},y)}{\partial \boldsymbol{\theta}} = vl(\sigma(\boldsymbol{\theta}^T\boldsymbol{x}) - y)\boldsymbol{x} + \lambda\boldsymbol{\theta}, \quad (22)$$

which is equivalent to the traditional LR learning with cross-entropy loss as in Eq. (16).

As a result, we discover that, under the assumption of (i) truthful bidding function and (ii) uniform market price distribution, our proposed learning models which directly optimize the profit-related utility are equivalent to the standard logistic regression with square error loss or cross entropy loss. Table 2 summarizes and provides a straightforward comparison among various model settings with the EU and RR loss. But in our settings, we adopt more reasonable bidding function and market price distribution, to achieve substantial improvement against the traditional regression loss methods.

3.6 Joint Optimization with Bidding Function

Once the CTR model has been trained, we can also use the newly updated model to fine-tune the bidding function such as the one in Eq. (12). The rational is that with the CTR estimator $f_{\boldsymbol{\theta}}(\boldsymbol{x})$, the distribution of predicted CTR changes, thus it is possible that the bidding function should also be updated to fit the new CTR distribution [32].

The click maximization framework in [32] limits its discussion by regarding the bid price as the upper bound cost. In this paper, we further extend the study by focusing on the strict second price cost. Another difference lies in that we consider the linear bidding function in Eq. (12) and intend to derive the optimal solution of parameter ϕ [3].

Specifically, once we fixed $f_{\boldsymbol{\theta}}(\boldsymbol{x})$, with the auction volume T and campaign budget B, we optimize ϕ in Eq. (12) as:

$$\begin{aligned}\arg\max_{\phi}\; & T\int_r\int_0^{\phi v r}(vr - z)p_z(z)dz\cdot p_r(r)dr \\ \text{s.t. }\; & T\int_r\int_0^{\phi v r} z p_z(z)dz\cdot p_r(r)dr = B,\end{aligned} \quad (23)$$

where to simplify our notation, we substitute $f_{\boldsymbol{\theta}}(\boldsymbol{x})$ with its predicted CTR variable r. The Lagrangian $\mathcal{L}(\phi,\mu) =$

$$T\int_r\int_0^{\phi v r}\left[vr - (\mu+1)z\right]p_z(z)dz\cdot p_r(r)dr + \mu B, \quad (24)$$

where μ is the Lagrangian multiplier. Taking the derivative equal to zero, we get that

$$\frac{\partial\mathcal{L}(\phi,\mu)}{\partial\phi} = 0 \;\Rightarrow\; \phi = \frac{1}{\mu+1}. \quad (25)$$

[3]As proven in [30], the theoretic optimal bidding function under the second price auction is linear w.r.t. CTR.

Algorithm 1 Joint optimization of CTR & bidding

Input: Training set D, learning rate α, total budget B
Output: Optimal $b()$ and $f_{\boldsymbol{\theta}}()$
1: Initially set parameter $\boldsymbol{\theta}$ and ϕ
2: **while** not converged **do**
3: (E-Step)
4: **for** each sample $(\boldsymbol{x}, y) \in D$ **do**
5: Calculate the gradient via Eq. (13) or (14)
6: Optimize $\boldsymbol{\theta}$ with gradient descent
7: **end for**
8: (M-Step)
9: Update bidding function $b(\cdot)$ via solving Eq. (26)
10: **end while**

To solve μ, we take the Lagrangian derivative w.r.t. to μ ant let it be zero, which obtains the constaint equation

$$T \int_r \int_0^{\frac{vr}{1+\mu}} z p_z(z) dz \; p_r(r) dr = B, \tag{26}$$

which normally has no analytic solution of μ except for some trivial implementation of $p_z(z)$ and $p_r(r)$. Fortunately, the numeric solution of μ is easy to find because the left part of the equation monotonously decreases against μ in the bidding function.

From Eq. (26), we find that the distribution of the predicted CTR $p_r(r)$ directly influences the optimal value of μ in the bidding function Eq. (25). It means that if we update the CTR estimation model $f_{\boldsymbol{\theta}}(\boldsymbol{x})$, then $p_r(r)$ will change accordingly, which in turn leads to the change of optimal μ. On the contrary, from Eqs. (13) and (14) we can see that if the bidding function $b(f_{\boldsymbol{\theta}}(\boldsymbol{x}))$ updates, the CTR estimation model $f_{\boldsymbol{\theta}}(\boldsymbol{x})$ will be updated too. Thus the CTR estimation and the bid function mutually influence each other.

As such, it is not ideal to simply optimize either of the CTR estimation model or the bidding function. We thus propose to jointly learn there two parts via an EM-like algorithm as given in Algorithm 1. Such iterations will surely get converged because each E or M-step will at least not reduce the objective value. Empirical study will be given in Section 4 to investigate the convergence properties and the quality of local minima of the proposed algorithms.

4. EXPERIMENTS

In this section, we first present the datasets and the experiment settings with evaluation metrics. Secondly we will present the user response prediction model performance and show the reason behind the improvement of our models. Thirdly, we also discuss the experimental results for jointly optimization framework and finally we will show our online A/B test experiments. We use an online learning paradigm, that is stochastic gradient descent, for training.

4.1 Datasets

We use two real-world datasets: iPinYou and YOYI, and provide repeatable offline empirical studies[4].

iPinYou is a leading DSP company in China. The iPinYou dataset[5] was released to promote the research on real-time bidding. The entire dataset contains 64.75M bid records including 19.5M impressions, 14.79K clicks and 16K CNY expense on 9 different campaigns over 10 days in 2013. The auctions during the last 3 days are set as test data while the rest as training data.

YOYI runs a major DSP focusing on multi-device display advertising in China. YOYI dataset[6] contains 402M impressions, 500K clicks and 428K CNY expense during 8 days in Jan. 2016. The first 7 days in the time sequence are set as the training data while the last 1 day is as the test data.

For the repeatable experiments, we focus on our study on iPinYou dataset. Our algorithms are further evaluated over the YOYI dataset for multi-device display advertising.

In real-time bidding, the training data contains much fewer positive samples than negative ones. Thus similar to [10], the negative down-sampling and the corresponding calibration methods are adopted in the experiment. The online A/B test is conducted on an operational real-time bidding platform run by YOYI.

4.2 Experiment Setup

Experiment Flow. We take the original impression history log as full volume bid request data. The data contains a list of bid record triples with user response (click) label, the corresponding market price and request features. We follow the previous work [32] for feature engineering and the whole experiment flow, which is as follows: the bid requests are received along with the time sequence, which is the same as the procedure that history log is generated. When received one request, our bid engine will decide the bid price to participate the real-time bidding auction. It wins if its bid price is higher than the market price, otherwise loses. On one hand, a truthful or linear bidding function is employed as our bidding strategy. On the other hand, we deploy different CTR estimation models to predict the user response probability, which then can be compared against each other. After bidding, the labelled clicks of the winning impressions will act as user feedback information. It is worth mentioning that this evaluation methodology works well for evaluating user response prediction and bid optimization [2, 32] and has been adopted in display advertising industry [16].

Budget Constraints. It is obvious that if our bid engine bids very high price each time, the cost and profit will stay the same as the original test log. Thus the budget constraints plays a key role in evaluation [32]. For the CTR estimation models, we only report for the test results without budget constraints since we care more about the prediction performance. For the joint optimization, we respectively run the evaluation test using 1/128, 1/64, 1/32, 1/16, 1/8, 1/4, 1/2 of the original total cost in the test log as the budget constraints.

4.3 Evaluation Measures

Since our objective is to improve the profit of a performance campaign and cut down the unnecessary cost in bidding, in our evaluation we measure **profit** and **ROI** w.r.t the corresponding cost in bidding phase. When the bid engine wins the auction, the corresponding market price will be added into the total *cost*. While the user response (click) is positive, we will take the campaign's value of this action as *return*. In our settings, this campaign value is set equal to eCPC in the history data log. The **profit** is regarded as the total gross profit ($\sum$ return $-\sum$ cost) for the whole test data auctions. **ROI** is another important measurement reflecting the cost-effectiveness of a bidding strategy. It can be regarded as a relatively orthogonal metric to auction volume and bid cost. We calculate **ROI** via profit/cost.

We also adopt commonly used **AUC** (Area Under ROC Curve)[7] and **RMSE** (Root Mean Squared Error) to mea-

[4]Repeatable experiment code: http://goo.gl/GzkCFQ.
[5]iPinYou Dataset link: http://goo.gl/9r8DtM.
[6]YOYI Dataset link: http://goo.gl/xaao4q.

[7]It has been shown that AUC is equal to the probability that a regressor correctly ranks a randomly chosen positive example higher than a randomly chosen negative one.

Table 3: Regression performances over campaigns. AUC: the higher, the better. RMSE: the smaller, the better.

	AUC				RMSE ($\times 10^{-2}$)			
iPinYou	SE	CE	EU	RR	SE	CE	EU	RR
1458	.948	.987	.987	.977	3.01	1.94	2.42	2.32
2259	.542	.692	.674	.691	2.01	1.77	1.76	1.79
2261	.490	.569	.622	.619	1.84	1.68	1.71	1.68
2821	.511	.620	.608	.639	2.56	2.43	2.39	2.46
2997	.543	.610	.606	.608	5.98	5.82	5.84	5.82
3358	.863	.974	.970	.980	3.07	2.47	3.32	2.67
3386	.593	.768	.761	.778	2.95	2.84	3.32	2.85
3427	.634	.976	.976	.960	2.78	2.20	2.61	2.34
3476	.575	.957	.954	.950	2.50	2.32	2.39	2.33
Average	.633	.794	.795	**.800**	2.97	**2.61**	2.86	2.69
YOYI	.882	.891	**.912**	**.912**	11.9	11.7	11.8	**11.6**

sure the accuracy of a regression model. We also take ad related metrics such as **eCPC**, cost per thousand impressions (**CPM**), **CTR**, and the **winning rate** to compare the bidding performance of the different prediction models.

4.4 Compared Settings

Test Settings without Budget Constraint. For the first part of our experiment, the unlimited budget is tested. All the CTR models are embedded with the same truthful bidding function. We compare 4 models in this part:

CE - The logistic regression model [10, 18] is widely used in many DSP platforms to make predictions of user feedback. This model takes cross entropy as its optimization objective and has the gradient as Eq. (16).

SE - This logistic regression model takes the squared loss as the objective function, which takes the gradient update as Eq. (15).

EU - Our proposed expected utility model, which takes the gradient update as Eq. (13).

RR - Our proposed risk-return model, which takes the gradient update as Eq. (14).

Test Settings with Budget Constraint. CTR learning and bid joint optimization with budget constraint is the second part of experiment. Here we test 4 solutions:

CELIN - As in [22, 32], the bid value is linearly proportional to the predicted CTR. We implement an LR model with a linear bidding strategy.

EUEM - This method combines our expected-utility model with bid optimization, which is described in Section 3.6 and trained with Algorithm 1.

RREM - This is our risk-return model with the consideration of a budget constraint, embedded in the joint optimization framework, also trained Algorithm 1.

RR_NOEM - The risk-return model without bid optimization is also considered to show the effectiveness of our joint optimization method. The truthful bidding function is used in this setting.

4.5 Accuracy of CTR Estimation

In this section, we compare the accuracy of the CTR estimation models, measured by AUC and RMSE. As our models are designed to optimize performance campaign revenue rather than user response accuracy, the evaluation here is to see whether our proposed solutions would still be able to achieve comparable performance against the conventional estimators that directly optimize the prediction accuracy. Table 3 shows the AUC and RMSE for each model over

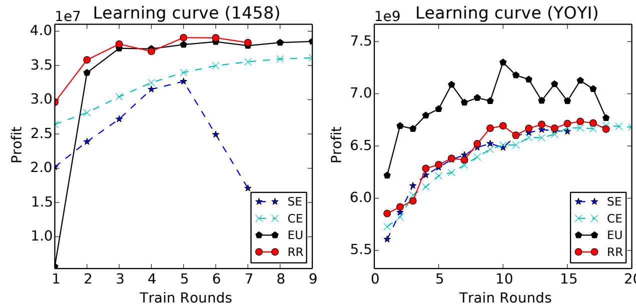

Figure 3: Training on iPinYou (left) and YOYI (right).

Table 4: Direct campaign profit over baselines.

	profit($\times 10^7$)		ROI	
iPinYou	SE	CE	SE	CE
1458	3.2	3.6	4.2	6.6
2259	-0.32	0.40	-0.080	0.18
2261	0.29	0.63	0.26	0.40
2821	0.11	0.08	0.21	0.023
2997	0.11	0.14	0.42	0.71
3358	1.76	2.4	5.4	5.2
3386	0.51	1.6	0.16	1.2
3427	0.33	2.9	0.11	3.4
3476	0.65	3.1	0.36	3.5
Average	0.74	**1.7**	1.2	**2.3**
YOYI	665.6	**669.5**	1.8	**1.9**

all campaigns. First, the baseline CE achieves better performance than the baseline SE on all campaigns, confirming the previous study that cross entropy as an objective naturally works on the binary classification problem with probabilistic predictions, whereas the squared error is more suitable for regression problems with a continuous target value [8]. Second, both our EU and RR models achieve similar or higher AUC values over the strong baseline CE model, while maintaining comparable RMSE performances. From our derivation in Section 3, we know that a key advantage of our EU model over the baseline SE model is that it considers the market price in the gradient updating. Here, we find that our EU model not only compensates the relatively weakness of the SE model, but also gains better in some campaigns, e.g. iPinYou campaign 2261 and YOYI. Moreover, the EU model achieves similar (sometimes better) performances compared with the CE model. Finally, we also observe that our RR model performs more stably in most campaigns and achieves higher AUC than other three models in most campaigns, e.g. iPinYou campaigns 2821, 3358, 3386 and YOYI, suggesting that combining the cross entry loss with the market price density is the best option.

4.6 Campaign Profit Optimization

As we have found in the previous section that our models have at least comparable performances for predicting CTR, we are now ready to examine the performance of profit optimization for each campaign in an unlimited budget setting (we will present the results under limited budget in Section 4.8). Figure 3 plots the obtained profit against the training rounds for the 4 models in both the iPinYou and YOYI datasets. The model will learn on the whole train set in one round. While the figures show the convergence of each estimation model, SE does not well generalize its CTR prediction to the profit optimization in iPinYou dataset. Compared to RR, EU's prediction focuses on medium-valued CTR cases, which is indeed the range with high volume of clicks in YOYI's market data, while RR focusing more on higher-valued cases. This results EU in winning more quality cases than RR.

We further examine the two baselines, SE and CE, with more details in Table 4. Both models achieve positive ROIs

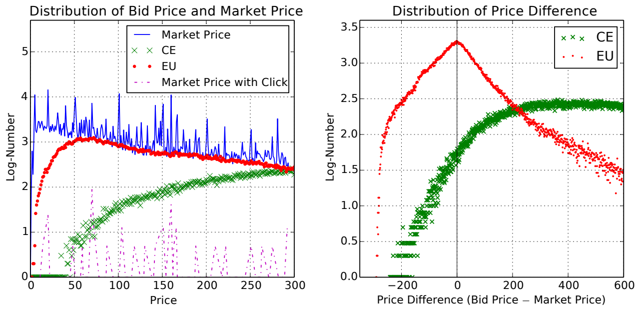

Figure 4: Analysis of bid price and market price distribution (iPinYou campaign 2259)

in almost all campaigns. And, in most campaigns except for iPinYou campaign 2821, the baseline CE outperforms the baseline SE in terms of the profit and ROI. This is consistent with our finding in the previous section that the CE model outperforms SE for CTR prediction accuracy.

We next pick up the best baseline CE model and use it as the base to compare the profit gain and ROI gain with our proposed EU and RR models, which are shown in Table 5. From the table, we can find that (i) Both the EU and the RR models consistently achieve higher profit than the CE baseline. Only in iPinYou campaign 3386, EU gains less profit. In average, our proposed models improve the profit about 71.2% for EU and 78.2% for RR, respectively. (ii) For the ROI metric, EU and RR get even higher overall improvements against the baseline CE. The average ROI gains are 202% for EU and 217% for RR, respectively. Those results suggest that our proposed models are much more cost-effective, meaning that they are capable of better budget allocation via bidding high value yet low cost impressions. (iii) the RR model is the best and in average, it gains 7.0% and 15% than EU in profit and ROI, respectively.

Finally, Table 6 provides other statistics to summarize the overall campaign performance for the 4 CTR estimation models. (i) CTR reflects the quality of the winning impressions, thus indicates whether a model would be able to target high CTR impressions. Our models, both EU and RR, outperform the baselines. (ii) CPM measures the cost of the winning impressions, thus indicating whether the model would be able to target impression economically. Our models achieve comparable CPM with baselines. (iii) Combining CTR and CPM leads to eCPC, which is the most relevant metric to profit and ROI. Both our models, EU and RR, pay less in most campaigns for each click, which explains why they gain much more profit and higher ROI. (iv) EU and RR lead to quite low winning rates to avoid over spending on low quality ad inventory and thus achieve relatively better profit and ROI performance.

4.7 Bidding Data Analysis

In this section, we further analyze the bidding data to gain more insights into why our models outperform the baselines. As we discussed in our formulation, a key advantage of our models is the introduction of the market price distribution and the utility of the bid to the learning of CTR model parameters. To understand the impact, in the left subfigure of Figure 4, we plot the distribution of bid values for our EU (similar to RR) model and the baseline CE model and compare them with the market price distribution and also the market price of the impressions that received clicks. We cut off the figure for price > 300 since the market price never goes beyond 300 in the dataset, and we will discuss the situations for high price soon later.

Firstly, we see that the bid prices generated from CE

Table 5: Campaign profit improvement over baseline CE.

iPinYou	Profit gain EU	Profit gain RR	ROI gain EU	ROI gain RR
1458	7.10%	9.00%	233%	267%
2259	81.6%	99.3%	233%	472%
2261	26.3%	31.1%	44.4%	91.2%
2821	573%	615%	1334%	943%
2997	5.00%	0.700%	-3.60%	-11.4%
3358	1.70%	6.70%	77.1%	77.7%
3386	-1.20%	2.50%	20.6%	58.3%
3427	5.50%	8.70%	52.0%	175%
3476	4.20%	8.60%	16.0%	91.1%
YOYI	9.04%	0.600%	14.8%	2.11%
Average	+71.2%	+78.2%	+202%	+217%

Table 6: Overall statistics in offline evaluation.

iPinYou	CTR ($\times 10^{-4}$) SE	CE	EU	RR	eCPC SE	CE	EU	RR
1458	34	33	59	190	17	11	4.3	3.4
2259	3.3	3.6	3.7	5.8	303	235	172	136
2261	2.4	2.7	3.0	2.8	234	212	188	168
2821	5.5	5.9	4.8	7.0	116	137	105	112
2997	31	25	26	27	9.8	8.2	8.3	8.6
3358	51	41	69	61	18	19	12	12
3386	7.8	11	13	15	90	48	43	36
3427	7.2	25	29	72.8	98	25	17.3	10
3476	6.4	16	17	33.1	111	34	30	20
Average	16	18	25	**46**	110	81	64	**57**
YOYI	16	18	**26**	24	12.9	12.4	**11.3**	12

iPinYou	CPM SE	CE	EU	RR	Win Rate SE	CE	EU	RR
1458	57	37	25	65	0.22	0.24	0.13	.041
2259	100	84	64	78	0.89	0.63	0.44	0.24
2261	57	56	56	46	0.55	0.81	0.71	0.67
2821	63	80	50	78	0.12	0.63	0.48	0.45
2997	30	20	21	22	0.55	0.63	0.65	0.63
3358	92	77	80	70	0.11	0.20	0.11	0.13
3386	71	54	55	55	0.82	0.45	0.36	0.29
3427	70	60	49	75	0.75	0.26	0.22	.082
3476	71	55	50	65	0.49	0.31	0.31	0.15
Average	68	58	**50**	62	**0.50**	0.46	0.38	0.30
YOYI	**20**	23	29	30	**0.36**	0.30	0.22	0.22

Table 7: High bid price (> 300) statistics

Model	Auctions	Budget	Largest Bid Price
Baseline CE	92.5%	14.0%	37,795
Our Model EU	10.3%	1.49%	13,901

deviate significantly from the market prices; a large portion of the bids from CE are very high, whereas the distribution (in log scale) of the market prices gently descends from 50 to 300, with its peak in the region between 0 and 30.

By contrast, our model EU nicely reduces the *difference* between the distributions of bid price and the market price by focusing the training on the cases that the bid is close to the market price (see the discussion in Section 3).

Moreover, considering the market price distribution of the impressions with clicks, we find that the bid distribution of EU fits it much better than that of CE, which means the bids from EU are more unlikely to miss high quality ad impressions than those from CE.

The right subfigure in Figure 4 further shows the distributions of the price *difference* between the bids (from EU and CE respectively) and the true market prices. We find that CE has a rather biased bidding strategy — a large portion of the bids are much higher than the corresponding market prices. For EU, on the contrary, the major proportion of the bids are in the "sensitive zone" where bid price is close to the market price. The peak is located at zero, which shows that EU can well model the market price distribution and perform more cost-effective bidding.

It is particularly important to control the over spending as some of RTB auctions are in fact the first price auction or with soft floor prices [29]. Table 7 gives the statistics related to high bid prices, where the bid value exceeds the highest market price 300 in our dataset. Specifically, for the

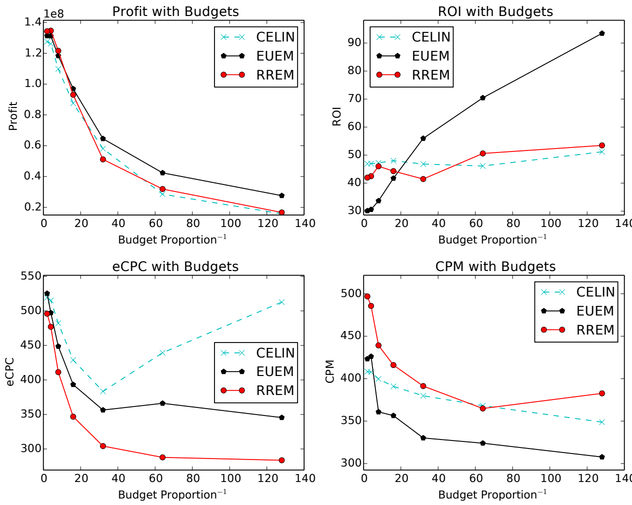

Figure 5: Performances with budgets on iPinYou.

Table 8: Profit improvements over CELIN (%).

Model	Budget Proportions^{-1} 2	4	8	16	32	64	128
EUEM	+2.9	+4.0	+7.8	+11	+11	+48	+75
RREM	+5.6	+6.7	+10	+6.3	-12	+12	+ 5.7

baseline CE there are 14.0% winning auctions with bid value exceeding 300. By contrast, our model EU substantially reduces the number of high bids and controls the high price auctions fewer than 1.5% in the whole bidding process.

4.8 Joint Optimization with Budgets

As formulated in Section 3.6, our model would be able to jointly optimize both the CTR estimation and bidding function by alternatively fixing one of them and optimizing the other. In this section, we evaluate our joint optimization models under budget constraints. We mainly compare three models: CELIN, EUEM and RREM as discussed in Section 4.4. And we set the test budget as 1/2, 1/4, 1/8, 1/16, 1/32, 1/64 and 1/128 of the original total cost in the history log respectively.

In Figure 5, we compare the overall performance for those three models over the tested campaigns. We find that in almost all the settings and the metrics, our proposed joint optimization models, EUEM and RREM, outperform CELIN. Table 8 further lists the detailed profit improvement over all budget settings.

From Figure 5, we also see that CELIN achieve higher ROI than our models when the budget relatively abundant (1/16 or more), but is instantly outperformed by our models in relatively limited budget settings (less than 1/16), which are more practical in real world. This phenomenon is reasonable since the bid optimization does not lead significant performance improvement in high budget settings, as also reported in [32], and our optimization goal is the direct profit instead of ROI. We also have the similar observations as those in Section 4.6 about CPM that RREM achieves higher CPM than EUEM because of the well control of the RR model for the risk and cost. The detailed achieved direct profit in each budget settting is listed in Table 9.

Figure 6 further shows the effectiveness of our joint optimization framework against the ones only optimizing CTR. RR_NOEM is the setting that uses only RR model with truthful bidding function. The result shows that RR_NOEM model plays well when the budget is abundant, but its performance drops largely with the budget is limited compared to the whole volume expense, which is the practical situation. And the profit achieved by the joint optimization models stay relatively high under those tighter budgets.

Table 9: Achieved direct profit ($\times 10^6$) with budgets^{-1}

iPinYou	CELIN 128	64	32	EUEM 128	64	32	RREM 128	64	32
1458	12	25	39	37	39	39	15	25	40
2259	.77	.98	2.2	.49	.71	2.2	1.0	1.5	2.5
2261	.07	.44	1.5	.66	.74	1.5	.67	1.0	2.4
2821	.027	.47	1.4	.59	1.7	1.6	.31	1.3	2.6
2997	.017	.033	.081	.059	.10	.35	.045	.23	.39
3358	6.7	11	22	16	22	24	5.1	7.8	16
3386	.90	1.5	2.5	1.8	2.1	5.8	2.1	3.8	7.0
3427	4.3	9.0	18	4.7	10	20	3.6	6.7	10
3476	3.0	4.3	10	2.2	4.5	8.0	3.7	9.4	10
Average	3.1	4.6	11	**7.1**	**9.0**	**11**	3.5	6.3	10

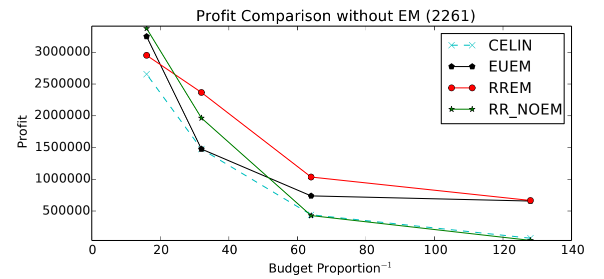

Figure 6: Joint optimization v.s. CTR optimization only.

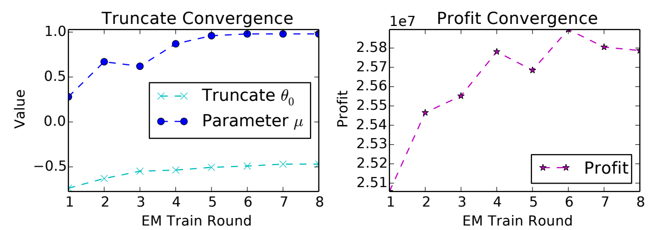

Figure 7: Demonstration for EM convergence.

We also experimentally illustrate the EM convergence of our joint optimization methods embedded with EU model in Figure 7. *Truncate* is the parameter value $\boldsymbol{\theta}_0$ of the weight for the constant additional feature in the CTR estimation function $f_{\boldsymbol{\theta}}(\boldsymbol{x})$. And μ is the parameter in the linear bidding function (25). The figure shows the value changes of *Truncate* and μ over EM-like training rounds. We see that our optimization converges in about Round 7 and the fluctuation of the observed parameters is small during the whole training stage, which shows the efficiency of our EM-alike algorithm.

4.9 Online Deployment and A/B Test

Our user response prediction models are deployed and tested in a live, commercial environment provided by YOYI PLUS (Programmatic Links Us) platform, which is a leading DSP in China. There are 4 deployed models: EU, RR, CE and FM, where the first three have been discussed in Section 4.4 and FM is a factorization machine model [21] with non-hierarchical feature engineering. To show the comparable performance of user response prediction, we set the same linear bidding function for all prediction models including baselines. The only difference is the embedded prediction model. The unit of money is CNY. We test over 10 campaigns during 25-26 January, 2016. The whole tested bid flow contains over 89 million auctions including 3.3 million impressions, 8,440 clicks and 1,403 CNY budget cost. All the models are trained on 7 days data which have more than 380 million impressions log and 52.5K CNY cost in total. The received bid requests are randomly selected to send to each model at each time according to the user cookie ID, while the chance controlled by the DSP platform for each

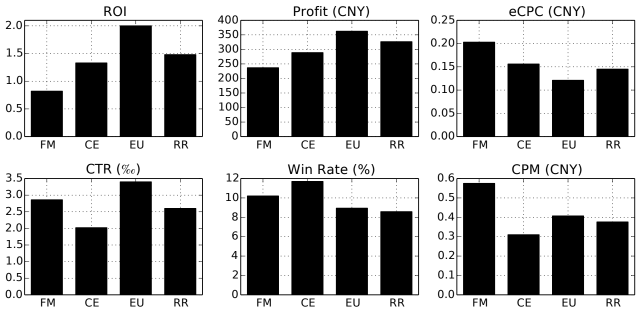

Figure 8: Online A/B testing results on YOYI PLUS.

model is set equal among all the 4 compared models. We set the same budget constraint for all deployed models. The overall result are presented in Figure 8.

From the comparison, we have the following conclusions: (i) EU and RR achieve higher profit and ROI than CE and FM. Specifically, EU has twice ROI as FM, and RR achieves more than 50% return of FM. EU gains 25.5% and 53.0% more profit than CE and FM respectively. (ii) eCPC consistently has inverse relationship with the trend of ROI. The online result also reflects this relationship: EU and RR have lower eCPC than other two baseline models. (iii) As for CTR, we find that EU achieves the highest CTR and RR also performs better than CE. Here FM has higher CTR than the CE model because it could learn feature interactions via the latent vector inner product [21]. However, FM obtains relatively less profit gain and ROI than CE, which shows that FM does not care enough about those auctions with high return value. (iv) EU and RR win fewer impressions but achieve more profit, which again indicates our proposed models are cost-effective. (v) Compared to the other 3 models, CE leads to higher winning rate with lower CPM. This suggests that CE allocates its budget on cheap ad inventory.

In sum, the online A/B testing results demonstrate the effectiveness of our proposed EU and RR training schemes in optimizing campaign profit. As for the difference between offline and online experimental results, it is reasonable because of the rapidly change of the market and variance among different campaigns.

5. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a novel user response model training scheme for directly optimizing campaign profit in RTB display advertising. Our mathematical derivations showed that the market price distribution and the bid utility contribute to the gradient updates as additional reweighting factors, which haven't been studied before. We tested our prediction models with other state-of-the-art estimation models under various budget settings. Up to 78.5% and 25.5% profit improvements were observed in the offline experiment and the online A/B testing, respectively, which verify the practical efficacy of our proposed training scheme of user response models.

A potential drawback of our current model is that although it works, we have a rather simple treatment for modeling the market price, that is obtained from the statistics of the history bidding logs. In the future, we plan to extend our model by considering more advanced bid landscape forecasting [27] and combining censored data learning [2] with the parameters jointly optimized in our framework.

6. REFERENCES

[1] D. Agarwal, R. Agrawal, R. Khanna, and N. Kota. Estimating rates of rare events with multiple hierarchies through scalable log-linear models. In *KDD*, 2010.

[2] K. Amin, M. Kearns, P. Key, and A. Schwaighofer. Budget optimization for sponsored search: Censored learning in MDPs. *UAI*, 2012.

[3] J. O. Berger. *Statistical decision theory and Bayesian analysis*. Springer Science & Business Media, 2013.

[4] O. Chapelle. Offline evaluation of response prediction in online advertising auctions. In *WWW*, 2015.

[5] Y. Chen, P. Berkhin, B. Anderson, and N. R. Devanur. Real-time bidding algorithms for performance-based display ad allocation. In *KDD*, 2011.

[6] Y. Cui, R. Zhang, W. Li, and J. Mao. Bid landscape forecasting in online ad exchange marketplace. In *KDD*, 2011.

[7] B. Edelman, M. Ostrovsky, and M. Schwarz. Internet advertising and the generalized second price auction: Selling billions of dollars worth of keywords. Technical report, National Bureau of Economic Research, 2005.

[8] J. Friedman, T. Hastie, R. Tibshirani, et al. Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *The annals of statistics*, 28(2):337–407, 2000.

[9] T. Graepel, J. Q. Candela, T. Borchert, and R. Herbrich. Web-scale bayesian click-through rate prediction for sponsored search advertising in microsoft's bing search engine. In *ICML*, pages 13–20, 2010.

[10] X. He, J. Pan, O. Jin, T. Xu, B. Liu, T. Xu, Y. Shi, A. Atallah, R. Herbrich, S. Bowers, et al. Practical lessons from predicting clicks on ads at facebook. In *KDD*, 2014.

[11] V. Krishna. *Auction theory*. Academic press, 2009.

[12] K. J. Lang, B. Moseley, and S. Vassilvitskii. Handling forecast errors while bidding for display advertising. In *WWW*.

[13] K.-c. Lee, A. Jalali, and A. Dasdan. Real time bid optimization with smooth budget delivery in online advertising. In *ADKDD*, 2013.

[14] K.-c. Lee, B. Orten, A. Dasdan, and W. Li. Estimating conversion rate in display advertising from past performance data. In *KDD*, pages 768–776. ACM, 2012.

[15] X. Li and D. Guan. Programmatic buying bidding strategies with win rate and winning price estimation in real time mobile advertising. In *Advances in Knowledge Discovery and Data Mining*, pages 447–460. 2014.

[16] H. Liao, L. Peng, Z. Liu, and X. Shen. ipinyou global rtb bidding algorithm competition dataset. In *KDD*, pages 1–6. ACM, 2014.

[17] Q. Liu, F. Yu, S. Wu, and L. Wang. A convolutional click prediction model. In *CIKM*, 2015.

[18] H. B. McMahan, G. Holt, D. Sculley, M. Young, D. Ebner, J. Grady, L. Nie, T. Phillips, E. Davydov, D. Golovin, et al. Ad click prediction: a view from the trenches. In *KDD*.

[19] A. K. Menon, K.-P. Chitrapura, S. Garg, D. Agarwal, and N. Kota. Response prediction using collaborative filtering with hierarchies and side-information. In *KDD*, 2011.

[20] S. Muthukrishnan. Ad exchanges: Research issues. In *Internet and network economics*. 2009.

[21] R. J. Oentaryo, E.-P. Lim, J.-W. Low, D. Lo, and M. Finegold. Predicting response in mobile advertising with hierarchical importance-aware factorization machine. In *WSDM*.

[22] C. Perlich, B. Dalessandro, R. Hook, O. Stitelman, T. Raeder, and F. Provost. Bid optimizing and inventory scoring in targeted online advertising. In *KDD*, 2012.

[23] M. Richardson, E. Dominowska, and R. Ragno. Predicting clicks: estimating the click-through rate for new ads. In *WWW*, pages 521–530. ACM, 2007.

[24] A.-P. Ta. Factorization machines with follow-the-regularized-leader for ctr prediction in display advertising. In *Big Data*, 2015.

[25] F. Vasile and D. Lefortier. Cost-sensitive learning for bidding in online advertising auctions. *NIPS Workshop: Machine Learning for eCommerce*, 2015.

[26] J. Wang and S. Yuan. Real-time bidding: A new frontier of computational advertising research. In *CIKM Tutorial*, 2013.

[27] W. C.-H. Wu, M.-Y. Yeh, and M.-S. Chen. Predicting winning price in real time bidding with censored data. In *KDD*, 2015.

[28] J. Xu, X. Shao, J. Ma, K.-c. Lee, H. Qi, and Q. Lu. Lift-based bidding in ad selection. In *AAAI*, 2016.

[29] S. Yuan, J. Wang, and X. Zhao. Real-time bidding for online advertising: measurement and analysis. In *ADKDD*, 2013.

[30] W. Zhang, K. Ren, and J. Wang. Optimal real-time bidding frameworks discussion. *arXiv:1602.01007*, 2016.

[31] W. Zhang and J. Wang. Statistical arbitrage mining for display advertising. In *KDD*, pages 1465–1474. ACM, 2015.

[32] W. Zhang, S. Yuan, and J. Wang. Optimal real-time bidding for display advertising. In *KDD*, 2014.