Efficient deep distribution network for bid shading in First-Price Auctions

Tian Zhou, Hao He, Shengjun Pan, Niklas Karlsson, Bharathbhusan Shetty, Brendan Kitts, Djordje Gligorijevic, Junwei Pan, San Gultekin, Tingyu Mao, Jianlong Zhang and Aaron Flores

Yahoo Research, Verizon Media
Sunnyvale, CA, USA
{tian.zhou,hao.he,alanpan,niklas.karlsson,bharatbs,brendan.kitts,djordje}@verizonmedia.com, jonaspan@tencent.com,{sgultekin,tingyu.mao,jianlong,aaron.flores}@verizonmedia.com

ABSTRACT
Since 2019, most ad advertising and sell-side platforms (SSPs), in the online advertising industry, shifted from second to first price auctions. Due to the fundamental difference between these auctions, demand-side platforms (DSPs) have had to update their bidding strategies to avoid bidding unnecessarily high and hence overpaying. Bid shading is the name given to techniques for reducing the price intended for second-price auctions, in order to balance cost and winning probability in a first-price auction setup. In this study, we introduce a novel deep neural network distribution-based approach in tackling both open (non-censored) and closed (censored) online first-price ad auctions. Offline and online A/B testing results show that our algorithm outperforms previous state-of-art algorithms in terms of both surplus and effective cost per action (eCPX) metrics. Furthermore, the algorithm is optimized in run-time and has been deployed into VerizonMedia DSP as production algorithm, serving hundreds of billions of bid requests per day. Online A/B test shows that advertiser’s ROI is improved by +2.4%, +2.4%, and +8.6% for impression based (CPM), click based (CPC), and conversion based (CPA) campaigns respectively.

CCS CONCEPTS
• Applied computing → Online auctions; • Information systems → Display advertising; • Computing methodologies → Machine learning algorithms.

KEYWORDS
Online bidding, bid shading, factorization machine, distribution learning

INTRODUCTION
In online advertising, inventories that are not directly sold are primarily auctioned programmatically in real-time bidding (RTB). Before 2018, second-price was the dominant form of auction for RTB, where the winner only needs to pay the second highest bid price. However, starting in 2017, all the major exchanges/SSPs including AppNexus, Index Exchange, OpenX, Rubicon Project, and Pubmatic, with the exception of Google AdX, were rolling out or testing first-price auctions, where the winner must pay whatever bid it submitted, in varying degrees [31]. Google transitioned to first-price auctions in 2019 [4]. Several motivations were behind the transition away from second-price auctions. Firstly, first price auctions provided greater transparency and accountability for bidders, since the bidder was always charged exactly what they offered [8, 12, 28, 32]. Secondly, the unmodified second price auctions proved to be incompatible with the widespread and popular practice of Header Bidding [18].

For demand-side platforms (DSPs), which are the bidders in the auctions, transitioning from second-price auctions to first-price auctions meant that bidding strategies would need to be dramatically adjusted. In second-price auctions, auction theory states that it is a dominant strategy for a bidder to bid truthfully [11], namely it is the optimal strategy for a DSP to compute the value of the inventory being auctioned and submit this value as the bid price, regardless of the other bidders’ behavior. However, for first-price auctions such a strategy would cause DSPs to overbid and thus lose money. This comes from the fundamental difference in the payment between second-price and first-price auctions.

Unlike in second-price auctions, in first-price auctions a DSP must incorporate other bidders’ behavior, more precisely its estimates of other bidders’ bid prices, into its own bidding strategy. If the competing bidders’ prices were known in advance, which is impossible in practice, the optimal bidding strategy would be to submit a bid price that is slightly higher than the highest competing bid price so as to win the auction with the lowest price possible. In reality, a DSP has to estimate the minimum winning price as best as it can, and lower its original bid price intended for second-price auction, i.e., shade the truthful value of the inventory, accordingly. This process is known as bid shading. Bid shading is relatively new to online advertising, but it has been used in auctions from other industries [6, 7, 10, 17].

The most important aspect of bid shading is a trade-off between the winning rate and (Return On Investment) ROI. The more the bid price is shaded, namely the lower the final bid price, the better the ROI if the bid is won. However, lower bid price also lead to lower...
The probability of winning a bid. The buyer or bidder surplus [17, 18] is the shaded amount, namely the difference between the bid price before shading and the final bid price if the bid is won. The surplus is 0 if the bid is lost. The quantitative objective of bid shading is thus to maximize the surplus, either directly or implicitly.

In the first price auction, an important piece of information is the minimum winning price, which is the highest competing bid, inclusive of the floor price, if provided. However, it’s up to an SSP to decide whether to provide the minimum winning price to the participating DSPs after the auction. Open (non-censored) first price auctions refer to the auctions where feedback including minimum winning price are shared to all participants regardless of auction outcome while in closed (censored) auction, only the win or loss feedback is available.

In this paper, we propose a deep distribution network to learn the distribution of minimum winning price for both censored and non-censored first price auctions, as well as for efficient search of the optimal bid price to maximize the profit in the real-time serving. The model has the flexibility of explicitly choosing the distributions of the minimum winning price and the structures of the network. Comprehensive experiments have been conducted and the model has been successfully deployed in one of the biggest DSPs in the world. We demonstrate the effectiveness of the algorithm by showing its performance lift compared to the existing models in offline and online setting.

To summarize, the main contributions of this study are the following:

- We propose a novel framework for bid shading that, to the best of our knowledge, is the unified distribution-based bid shading method can be applied for both censored and non-censored first-price auctions.
- We mathematically prove that an efficient search algorithm can be used to find the optimal bid price that maximizes surplus given the distributions. This allows the model to be deployed online under strict latency constraints.
- We implement and test the proposed algorithm in online and offline settings. The key metrics results show +9.7% and +14.3% lift for offline and online A/B tests respectively, compared to the existing models. Moreover, advertiser’s ROI are improved by +2.4%, +2.4%, +8.6% for impression based (CPM), click based (CPC), and conversion based (CPA) campaigns respectively.

The rest of the paper is organized as follows: In Section 2 we discuss related work in bid shading and online auctions. In Section 3 we describe the formulation of our bid shading algorithms, including details on how we train models to estimate the distributions of minimum winning price and how to search for the optimal bid price efficiently on both censored and non-censored first-price auctions. In Section 4, we will show the comprehensive experiments of different choices of distributions and network structures compared to the current state-of-art bid shading algorithms for both censored and non-censored first price auctions. In Section 5 we briefly introduce the online deployment of deep distribution network in the serving system of our DSP and show the online ROI improvements compared to productions. Finally we conclude in Section 6.

## 2 RELATED WORK

Bid Shading is a buy-side activity, where the buyer has a private value which is the most they can offer, and attempts to reduce the price they are offering, without lowering it so far that they lose the auction. Bid shading is ubiquitous in repeated first price auctions, and is expected by auction theory [10, 17, 35]. Bid Shading shares characteristics with the Seller’s (Reserve) Price Optimization Problem [5, 20, 22, 23, 29, 30], where sellers have a good with a manufacturing cost, and their task is to set their price above this cost to maximize their profit. In both the Seller pricing problem, and the Buyer Bid shading problem, each entity is trying to maximize profit. However buyers are trying to minimize their offer and receive reward in proportion to the lowness of their price, where-as sellers are trying to maximize their reserve price. A variety of constraints also exist in the Buyer problem that are unique: (i) Buyers need to predict the bidding behavior of competing buyers on each auction, leading to strategic considerations. (ii) Buyer Feedback is constrained in systematic ways, with censored and uncensored information provided. (iii) Buyers need to find a bid price for billions of auctions, each of which has combinatorial aspects; where-as sellers generally have a set of fixed inventory [1, 3]. For these reasons, most authors talk about the Bidding problem as a Buyer specific activity, distinct from Seller reserve pricing, and we take the same approach in this paper.

There have been two general approaches for bid shading, depending on whether the minimum winning price is provided or censored. The first assumes that the minimum winning price is provided, and builds a machine learning algorithm to predict the optimal shading factor - the ratio of the bid price before shading over the minimum winning price. For instance, Logistic Regression (LogReg) and Factorization Machines (qFwFM) [13] have been used previously to predict the optimal shading factor, using an asymmetric loss function which penalizes losses. The drawback of these approaches is that they learn from the winnable bids only [13], ignoring a large portion of available data. This paper proposes an algorithm for modeling distributions over the entire bidding landscape which allows for learning from both won and lost auctions.

The other general approach tries to estimate the distribution of the minimum winning price at segment level, and then finds the optimal bid price by maximizing the expected surplus with respect to the estimated distribution. The NonLinear algorithm in [19] attempts to estimate the distribution using a non-linear approximation on a predefined segment. The main drawback of this method is that the distribution is estimated separately, and thus cross-segment information is not utilized. Furthermore, the segments must be explicitly defined and small segments must be manually grouped together. The NonLinear algorithm was outperformed by LogReg and qFwFM when the minimum winning price is available. The recent WinRate model from [25] takes a similar approach but approximates the distributions, implicitly with logistic, for all segments simultaneously. The WinRate algorithm outperforms both LogReg and NonLinear while performs comparably to qFwFM. The main drawback of the WinRate model is that it doesn’t utilize the minimum winning price information when available.
Outside of the problem of bid shading, there has been some related research trying to characterize winning prices on auctions. For example, in [33, 34] the authors first estimate the winning price distribution, and then use it to do point-wise prediction of the winning price via a mixture model. The distribution implicitly assumes that the winning price follows logistic distributions, while a log-logistic distributions are used in [25]. Additionally, there has been prior work estimating the bidding landscape in second-price auctions [33, 34], in which the minimum winning price feedback is only available while winning the auction. These approaches would need to be extended to work under first-price auctions, and the problem of surplus maximization.

Sell-Side Platforms are motivated to provide bid shading services to the bidders, especially during the transition period from second-price to first-price auctions. Such services include Bid Translation Service from Google AdX [14], Estimated Market Rate from Rubicon Project [27], and Bid Price Optimization system from AppNexus [2]. However, these services are not designed to optimize the surplus for DSPs or their advertisers as these services are designed to shade private value bid down to a price at which the bidders continue to win at a similar rate, but with a slightly lower price. Moreover, these services are rather a transition tool in helping DSP’s transition to first-price auctions, and many of them are being deprecated. For example, AdX deprecated its Bid Translation Service in May 2020 [16].

3 ALGORITHMS

For the reader’s convenience, we list some notations that will be used throughout the paper.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>V</td>
<td>bid price before shading (true value of the ad opportunity)</td>
</tr>
<tr>
<td>S</td>
<td>surplus</td>
</tr>
<tr>
<td>s</td>
<td>expected surplus: ( E[S] )</td>
</tr>
<tr>
<td>b</td>
<td>bid price</td>
</tr>
<tr>
<td>( \hat{b} )</td>
<td>minimum winning price</td>
</tr>
<tr>
<td>( b^* )</td>
<td>optimal bid price that maximizes the expected surplus ( s )</td>
</tr>
<tr>
<td>D</td>
<td>distribution of the minimum winning price</td>
</tr>
<tr>
<td>F(b)</td>
<td>Cumulative Distribution Function (CDF) of distribution ( D ), i.e., winning probability at bid price ( b )</td>
</tr>
<tr>
<td>f(b)</td>
<td>Probability Density Function (PDF) of distribution ( D ): ( F'(b) )</td>
</tr>
<tr>
<td>x</td>
<td>input feature vector that defines a segment</td>
</tr>
<tr>
<td>( \alpha )</td>
<td>parameters of a distribution</td>
</tr>
</tbody>
</table>

First of all, we define surplus mathematically. Upon receiving a bid request for first-price auction, its value \( V \) is estimated based on its event rate, such as click-through rate (CTR) or conversion rate (CVR), and a campaign level price control signal that helps to pacing campaign budget smoothly across the flight. Let \( b \) be the bid price to be submitted, and \( \hat{b} \) be the minimum winning price. \( \mathbb{I}(b > \hat{b}) \) equals 1 if \( b > \hat{b} \) and 0 otherwise, which indicates with bid price \( b \) whether we win the auction. Then the surplus is defined as

\[
S(b; V, \hat{b}) = (V - b) \mathbb{I}(b > \hat{b}) = \begin{cases} V - b, & \text{if } b > \hat{b}, \\ 0, & \text{otherwise}. \end{cases}
\]

Let \( x = (x_1, x_2, \ldots, x_k) \) be the input feature vector derived from the current publisher and user attributes, such as top level domain, sub-domain, layout, day of week, etc. We calculate the optimal bid price \( b^* \) given the current input feature vector in two steps:

**Distribution Estimation** First we build a machine learning model to approximate the conditional distribution \( D_{\hat{b}|x} \) of the highest competing bid price \( \hat{b} \) by modeling its PDF or CDF.

**Surplus Maximization** Then we find the bid price \( b = b^* \) that maximizes the expected surplus \( \mathbb{E}_{\hat{b}|x}[S] \):

\[
b^* = \arg \max_{b \in (0, V)} \mathbb{E}_{\hat{b}|x}[S(b; V, \hat{b})] = \arg \max_{b \in (0, V)} \mathbb{E}_{\hat{b}|x}[(V - b) \mathbb{I}(b > \hat{b})] = \arg \max_{b \in (0, V)} (V - b) \Pr(\hat{b} < b \mid x). \tag{2}
\]

In the following sections, we describe in more details how the distribution estimation and surplus maximization are conducted.

3.1 Inference of Distribution

Conditioning on input feature vector \( x \) of a bid request, to find the optimal bid price \( b^* \) that maximizes the expected surplus, modelling of winning probability \( \Pr(\hat{b} < b; \hat{b} \mid x) \) for any intended bid price \( b \) is the key. We assume that for given \( x \), the minimum winning price follows a conditional distribution \( D_{\hat{b}|x} \). We further assume that each \( \hat{b} \) is drawn independently from a probability distribution \( D_{\hat{b}|\alpha(x)} \) that belongs to a family of known distributions, where \( \alpha = (\alpha_1, \alpha_2, \ldots, \alpha_m) \) is its \( m \)-parameters vector. Note that the parameter vector \( \alpha \) is a function of \( x \), namely, bid samples with the same input feature vector follow the same distribution. We will discuss details about different distributions at greater length in Section 3.3.

3.1.1 Minimum winning price is provided by SSPs. In non-censored first-price auctions, the minimum winning price \( \hat{b} \) is provided by the SSP after the auction regardless of the outcome. We build a machine learning model to estimate the distribution of \( D_{\hat{b}|x} \). Let \( \{(x^{(i)}, \hat{b}_i)\}_{i=1}^n \) be the training data set of bid samples, where \( x^{(i)} \) is the input vector, \( \hat{b}_i \) the minimum winning price provided by the SSP, and \( n \) is the total number of training samples. Let \( f(\hat{b}; \alpha(x^{(i)})) \) be the corresponding PDF. Then we estimate \( \alpha \) using maximum likelihood estimation (MLE):

\[
\max_{\alpha} \sum_{i=1}^{n} \log f(\hat{b}_i; \alpha(x^{(i)})), \tag{3}
\]

where \( D \) is a predetermined distribution family. The model structure for maximizing the log-likelihood is illustrated in Figure 1. Note that compared to \( x \), \( \alpha \) is of much lower dimension, namely \( k \gg m \).

3.1.2 Minimum winning price is censored by SSPs. If the SSP does not provide the minimum winning price after the auction, the previous MLE approach will not work. In this case, we use an approach similar to that in [25] to estimate the CDF of \( D_{\hat{b}|x} \), adding the flexibility of choosing the distribution family. Let \( \{(x^{(i)}, b_i, \ell_i)\}_{i=1}^n \) be the training data set of bid samples, where \( x^{(i)} \) is the input feature vector, \( b_i \) is the submitted bid price, and \( \ell_i \in \{0, 1\} \) indicates if the
we formulate it as a prediction problem. More precisely, we estimate the expected surplus function with respect to bid price, let $D$ be expected surplus function with respect to $b$.

The model structure for CDF estimation when the minimum winning price is censored by SSPs is as follows. Given $x$ and the bid price before shading $V$, the objective is to solve the following maximization problem:

$$\max_{b \in (0, V)} s(b; V, x).$$

If the underlying distribution $D_{b|x}$ results in a surplus function that has a unique local extremum (maximum or minimum), we can adapt the golden section search algorithm [21] shown in Algorithm 3.1 which converges in logarithmic time. It is a more numerically stable approach for most distributions, especially for gamma distribution and log-normal distribution, since no calculation of gradient is needed. The golden section search is more versatile and robust than other algorithms that require the calculation of gradients, which makes it more suitable for online implementation.

**Algorithm 3.1 Golden Section Search for Surplus Maximization**

**Require:**
1. $V$: estimated value of the current ad opportunity
2. $s(b)$: expected surplus function;
3. $\epsilon > 0$: minimum valid interval length
4. $N$: maximum number of search steps

**Ensure:**
1. $\beta > 0, V > 0$, and $s(b)$ has exactly one local maximum and no local minimum in $(0, V)$.
2. $b_{\min}, b_{\max} \leftarrow 0, V$
3. $gr \leftarrow (\sqrt{5} + 1)/2$
4. $x_1 \leftarrow b_{\max} - (b_{\max} - b_{\min})/gr$
5. $x_2 \leftarrow b_{\min} + (b_{\max} - b_{\min})/gr$
6. for $i = 1, 2, \ldots, N$ do:
7. if $s(x_1) > s(x_2)$ then $b_{\max} \leftarrow x_2$ else $b_{\min} \leftarrow x_1$
8. if $b_{\max} - b_{\min} < \epsilon$ then break
9. $x_1 \leftarrow b_{\max} - (b_{\max} - b_{\min})/gr$
10. $x_2 \leftarrow b_{\min} + (b_{\max} - b_{\min})/gr$
11. end for
12. return $(b_{\min} + b_{\max})/2$

**3.3 Distribution Families**

There are some constraints in choosing the distribution of minimum winning price to make economic sense. For example, its PDF should have a support between 0 and positive infinity. We mainly focus on truncated-normal, exponential, gamma, and log-normal distribution families and show that the surplus functions for them have a unique local extremum, so that Algorithm 3.1 is applicable to them all. We first introduce a few notations. Assuming that the minimum winning price $\hat{b}$ follows a probability distribution $D$ with CDF $F(b)$ and PDF $f(b)$, then expected surplus and its first and second derivatives can be calculated as

$$s(b) = (V - b)Pr(\hat{b} < b | \hat{b} < V),$$

$$s'(b) = (V - b)f(b) - F(b),$$

$$s''(b) = (V - b)f'(b) - 2f(b).$$

For simplicity, we don’t explicitly write the above as functions of input feature vector $x$ and bid price before shading $V$ here.

We first show that for all truncated-normal, exponential, gamma, and log-normal distributions, $s''(b)$ has at most one root in $(0, V)$.
Throughout this paper, we use the truncated-normal distribution.

### Lemma 1. For any truncated-normal, exponential, gamma, or log-normal distribution of the minimum winning price, $s''(b)$ has at most one root in $(0, V)$ and $s''(V) < 0$. Thus, for all these four distributions, the corresponding surplus function has one global maximum.

#### Proof. From Lemma 1, for any truncated-normal, exponential, gamma, or log-normal distribution, $s''(b)$ has at most one root in $(0, V)$ and $s''(V) < 0$. Then only one of the following two cases could happen:

- $s''(b) < 0$ for all $b \in (0, V)$, in which case $s(b)$ is concave and hence has at most one local maximum and no local minimum.
- $s''(b)$ has a unique root $b_0 \in (0, V)$ such that $s''(b) > 0$ for all $b \in (0, b_0)$ and $s''(b) < 0$ for all $b \in (b_0, V)$, in which case $s(b)$ is convex in $(0, b_0)$ but concave in $(b_0, V)$.

In both cases, $s(b)$ has at most one local maximum and no local minimum. Further, since $s(0) = S(V) = 0$ and $s(b) > 0$ for all $b \in (0, V)$, $s(b)$ must have one global maximum. □

### Offline Experiments

In this section, we present comprehensive offline experiments on our DSP private bidding dataset. The following questions would be answered in the following sub-sections:

- **Q1:** Does lower log-loss or better distribution fit results in a higher surplus?
- **Q2:** How much is the surplus lift when minimum winning price is available in training the deep distribution network compared with when minimum winning price is not available?
- **Q3:** Will performance be improved by using more powerful network structures (deepFM etc.) that capture high-order feature interactions compared to logistic regression or Factorization Machines (FM)?

#### 4.1 Dataset

The dataset we use for offline experiments is Verizon Media DSP private bidding dataset on Adx exchanges. We extracted 12 fields through feature engineering, including exchange id, top level domain of the ad opportunity, sub domain, layout of the ad, position of the ad, device type, name of app, publisher id of the ad request, country, user local hour of the day, user local day of the week, user local time of the week, user new. There are billions of records, with millions of active features and minimum winning price available. We use 7 day’s data to train the prediction model and use 1 day’s data to test. Notice that we train two types of models separately to simulate censored and non-censored first-price auctions. For the pdf estimation model which introduced in Fig. 1, minimum winning price are used as labels, while for the cdf estimation model in Fig. 2, the binary win (0/1) is used as label. We used the bid requests in the past 7 days for training to mitigate impact of the day of week pattern. Log-loss and surplus are the main two metrics we use in the offline experiment.

#### 4.2 Minimum winning price provided by SSPs

The production bid shading algorithms for non-censored first-price auctions is Factorization Machine based point estimation algorithm. It uses Field-weighted Factorization Machine (FwFM) as the model structure and learns the optimal shading factor for each bid request.
with an asymmetric loss function (penalize more when losing the bid) [13]. To have a fair comparison with the baseline, we use same model structure: FwFM as the model structure in Figure 1, and conduct experiments of different distributions we introduced in Section 3.3. For surplus metric, we only show the percentage of lift compared to the production model for privacy reason. Table 1 summarizes the results and we do observe the correlation between log-loss and surplus, in the sense that lower log-loss results in higher surplus. Among all the distributions, log-normal has the best performance with 9.7% surplus lift compared to the current production algorithm possibly due to its capability to better model the long-tail distribution of the minimum winning price.

Table 1: Performance of different distributions compared to production FwFM algorithm where highest competing bid price is not censored by the SSP

<table>
<thead>
<tr>
<th>Model</th>
<th>Log loss</th>
<th>Surplus Lift Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>truncated-normal</td>
<td>0.87</td>
<td>1.45%</td>
</tr>
<tr>
<td>exponential</td>
<td>0.68</td>
<td>5.12%</td>
</tr>
<tr>
<td>gamma</td>
<td>0.58</td>
<td>6.85%</td>
</tr>
<tr>
<td>log-normal</td>
<td>0.56</td>
<td>9.65%</td>
</tr>
</tbody>
</table>

4.3 Minimum winning price not provided by SSPs

To simulate the censored first-price auctions where minimum winning price feedback is not available. We train the deep distribution network on the same dataset while not using minimum winning price information during training. The loss function is defined in Section 3.1.2. The production algorithm on censored SSPs is win-rate distribution algorithm, in which winning probability function is estimated as a sigmoid function [25].

Table 2: Performance of different distributions compared to production win-rate distribution algorithm

<table>
<thead>
<tr>
<th>Model</th>
<th>Log loss</th>
<th>Surplus Lift Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>truncated-normal</td>
<td>1.14</td>
<td>0.10%</td>
</tr>
<tr>
<td>exponential</td>
<td>0.89</td>
<td>3.29%</td>
</tr>
<tr>
<td>gamma</td>
<td>0.72</td>
<td>3.58%</td>
</tr>
<tr>
<td>log-normal</td>
<td>0.68</td>
<td>5.32%</td>
</tr>
</tbody>
</table>

4.4 Network structure comparison

In this subsection, we present the results on comprehensive experiments on applying well-known click-through rate prediction models like FM, FwFM, deepFM, wide & deep etc., to deep distribution network, as the deep distribution network structure, in learning the distribution of minimum winning price. [9, 15, 24, 26]. The results are shown in Table 3 which answers Q3. It can be seen that deep models, like deepFM, wide & deep, which capture high-order feature interaction are able to learn a better distribution with relatively low log-loss, and thus have a better surplus compared to linear model LR, and shallow models FM, FwFM.

Table 3: Performance of different network structures with log-normal as pre-defined distribution

<table>
<thead>
<tr>
<th>network structure</th>
<th>Log loss</th>
<th>Surplus Lift Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>LogReg</td>
<td>0.718</td>
<td>/</td>
</tr>
<tr>
<td>FM</td>
<td>0.569</td>
<td>4.52%</td>
</tr>
<tr>
<td>FwFM</td>
<td>0.558</td>
<td>4.70%</td>
</tr>
<tr>
<td>wide &amp; deep</td>
<td>0.522</td>
<td>6.36%</td>
</tr>
<tr>
<td>deepFM</td>
<td>0.521</td>
<td>7.10%</td>
</tr>
</tbody>
</table>

Based on the offline experiments results and online latency constraint, we eventually decide to use FwFM as the network structure and log-normal distribution for online experiments. However, we show the potential of more complex deep network structure, if the online latency requirements can be satisfied.

5 ONLINE EXPERIMENTS

In this section, we will briefly introduce the real-time bid shading serving module we implemented, which serves billions of bid requests per day in one of the world’s largest DSP. The online experimental results, including Return on Investment (ROI) for advertisers, are also presented, which further prove that deep distribution network outperforms other existing bid shading algorithms in literature.

5.1 Bidding System Overview

Our DSP is a single platform that brings programmatic, premium, and its native marketplace inventory, formats, targeting and measurement together. We provide an overview of the bid shading aspects of the system, as illustrated in Figure 3.

When a user visits a web page with an ad opportunity, an ad request is sent to the SSP responsible for selling it. The SSP then packages available user and page information into bid requests and send them to multiple DSPs for auction. Upon receiving a bid request, within hundreds of milliseconds our bidding system goes through various stages such as fetching user profiles, ads targeting,
5.2 Online Evaluation Metrics

The most important key performance metrics for campaign delivery that the DSP system seeks to optimize towards are CPM (Cost per thousand impressions), CPC (Cost per click) and CPA (Cost per action). For our online A/B test experiments of the bid shading algorithms described in this paper, we mainly focus on campaigns with one of those 3 optimization goal types, which cover more than 80% of total DSP revenue. For a given campaign spend budget, the DSP optimization system seeks to minimize the campaign’s CPM, CPC or CPA as indicated by its optimization goal. For convenience, we use the term effective cost per event (eCPX) to denote either CPM, CPC or CPA.

Since optimization goals are defined at the campaign level, there is a need to define metrics to measure impact of algorithm improvements (in our case bid shading) across multiple campaigns. Simple aggregations of events (e.g. actions) and cost across campaigns to define a simple aggregated eCPX metric are not good metrics, since a few campaigns can dominate such metrics if it turns out they have orders of magnitude more events than the rest of the campaigns, even though their cost (campaign spend) is as high as any other campaign’s spend. Moreover, it’s not fair to compare two algorithms when their spends are different since the one with more spend tends to have higher eCPX. Thus, we are going to introduce two novel online DSP business-related metrics: campaign level eCPX statistics, and Bidder Performance Index (BPI), which can be used to measure improvements in campaign performance and ROI for our advertisers.

5.2.1 Campaign Level eCPX statistics. A natural way to put together eCPX metrics across campaigns is to generate statistics on improvements measured at the campaign level, i.e. based on the eCPX of control and test buckets for each campaign. To avoid the asymmetry implied by the usual percentage difference in eCPX between control and test, we define the log of eCPX ratio for each campaign as the following.

$$r_{ecpx} = \log\left(\frac{eCPX_{Test}}{eCPX_{Control}}\right).$$

For a campaign that has similar spend in both control and test bucket, if $r_{ecpx} < 0$, then the test algorithm outperforms the production model. This provides a metric of improvement (or deterioration) in eCPX that is symmetric in the sense that the absolute value of the metric remains unchanged if control and test roles are interchanged. A histogram depicting the distribution of $r_{ecpx}$ is shown in Fig 4 for CPA goal type campaigns. Based on the $r_{ecpx}$ distribution, we can measure that 71.5% of campaigns have better eCPX, and the median and mean of $r_{ecpx}$ are, respectively, $-0.035$ and $-0.038$. Since campaigns can have very different budget and spend, the weighted histogram of $r_{ecpx}$ is shown in Fig 5 where the weights are the spend of each campaign.

5.2.2 Bidder Performance Index (BPI). Minimizing eCPX for a given spend amount is equivalent to maximizing value or Return to advertisers under the same amount of spend (Cost), where Return to advertisers is the monetary value of the events (impressions, clicks or actions) driven by the campaign. This motivates the so-called BPI metric, defined as:

$$\frac{(Return_{test} - Cost_{test}) - (Return_{control} - Cost_{control})}{Cost_{control}}$$

where Return and Cost can be aggregated across campaigns. The numerator shows the extra Return the new algorithm brings to the advertiser compensated by the extra cost it may also incur. Notice that it can be negative if the test algorithm is no better than the control one. The BPI metric is an aggregation-based type of metric that is less prone to be dominated by campaigns with very large number of events as explained before.

5.3 Online A/B test results

After rolling out the deep distribution network algorithm with FwFM network structure and log-normal distribution on Adx, one of the largest SSPs, we were able to monitor its online performance by maintaining a percentage of traffic that was randomly allocated to each algorithm. There are thousands of campaigns running everyday in our DSP, the summary of these campaigns under our A/B test environment during 3 consecutive days is shown in Table 4.

It can be seen that for all goal types, the median of $r_{ecpx}$ are negative values, showing a overall better performance of our proposed algorithm. The weighted median, where the weights are the spend of each campaign, are also negative across all goal types. A majority of campaigns have a better eCPX performance on test bucket as can be seen from the third row of better campaign percentages. And
These campaigns take up to 70% to 80% of the total spend. In all, from the online A/B test results, it can be shown that our proposed bid shading algorithm generates a significantly better performance than the current model in production.

<table>
<thead>
<tr>
<th>Metric</th>
<th>CPA</th>
<th>CPC</th>
<th>CPM</th>
</tr>
</thead>
<tbody>
<tr>
<td>median of $\text{re}_{\text{ecpx}}$</td>
<td>-3.5%</td>
<td>-0.9%</td>
<td>-6.2%</td>
</tr>
<tr>
<td>median of spend weighted $\text{re}_{\text{ecpx}}$</td>
<td>-4.1%</td>
<td>-5.4%</td>
<td>-5.7%</td>
</tr>
<tr>
<td>better campaigns</td>
<td>62.3%</td>
<td>54.5%</td>
<td>81.6%</td>
</tr>
<tr>
<td>better spend</td>
<td>71.5%</td>
<td>80.6%</td>
<td>71.8%</td>
</tr>
<tr>
<td>BPI</td>
<td>+8.57%</td>
<td>+2.35%</td>
<td>+2.35%</td>
</tr>
</tbody>
</table>

6 CONCLUSIONS

In this work, we propose the deep distribution network for bid shading, which can be applied for both censored and non-censored first-price auctions. The parametric conditional distribution of minimum winning price is learnt through a FwFM network based on selected features of the bid request. For several well-known distributions, we proved that the resulting surplus function has a unique local maximum. Based on such property, an efficient golden-section search algorithm was applied at the real-time in finding the optimal bid price that maximizes expected surplus. In online experiments, the proposed model out-performed existing state of the art bid shading algorithm on both censored and non-censored scenarios with respectively 9.7% and 5.3% surplus lift. Online A/B test showed that the proposed algorithm increases surplus by +14.3% and brings +2.4%, +2.4%, +8.6% ROI lift for impression based (CPM), click based (CPC), and conversion based (CPA) campaigns respectively. The deep distribution network has been successfully deployed in one of the biggest DSPs in the world, serving billions of bid requests every day. Another major advantage of our framework is that the overall structure with a deep neural network and a parametric distribution output layer, can be easily generalized. In the future, we will continue exploring more powerful and run-time efficient neural network structures, combining with other single or multi mode distributions for further improved performance.

REFERENCES

[18] B. Hovaness. 2018. Sold for more than you should have paid. https://www.hearts-science.com/sold-for-more-than-you-should-have-paid/


