On Designing a Two-stage Auction for Online Advertising

Yiqing Wang[1], Xiangyu Liu[2], Zhenzhe Zheng[1]*, Zhilin Zhang[2], Miao Xu[2], Chuan Yu[2] and Fan Wu[1]

[1]Shanghai Jiao Tong University, China   [2]Alibaba Group, China

{wangyiqing_2015, zhengzhenzhe}@sjtu.edu.cn, fwu@cs.sjtu.edu.cn

{qilin.lxy, zhangzhilin.pt, xumiao.xm, yuchuan.yc}@alibaba-inc.com

ABSTRACT

For the scalability of industrial online advertising systems, a two-stage auction architecture is widely used to enable efficient ad allocation on a large set of corpus within a limited response time. The current deployed two-stage ad auction usually retrieves an ad subset by a coarse ad quality metric in a pre-auction stage, and then determines the auction outcome by a refined metric in the subsequent stage. However, this simple and greedy solution suffers from performance degradation, as it regards the decision in each stage separately, leading to an improper ad selection metric for the pre-auction stage. In this work, we explicitly investigate the relation between the coarse and refined ad quality metrics, and design a two-stage ad auction by taking the decision interaction between the two stages into account. We decouple the design of the two-stage auction by solving a stochastic subset selection problem in the pre-auction stage and conducting a general second price (GSP) auction in the second stage. We demonstrate that this decouple still preserves the *incentive compatibility* of the auction mechanism. As the proposed formulation of the pre-auction stage is an NP-hard problem, we propose a scalable approximation solution by defining a new subset selection metric, namely *Pre-Auction Score (PAS)*. Experiment results on both public and industrial dataset demonstrate the improvement on social welfare and revenue of the proposed two-stage ad auction, than the intuitive greedy two-stage auction and other baselines.

CCS CONCEPTS

• **Information systems → Computational advertising**.

KEYWORDS

online advertising, ad auction, two-stage auction

1 INTRODUCTION

Online advertising is the major sources of revenue for Internet industry [14]. Modern online advertising platforms conduct ad allocation by running ad auction mechanisms in real-time. To achieve effectiveness and efficiency in ad allocation, the ad auction mechanisms jointly consider the bids from advertisers as well as the quality of displaying ads to users. For example, in the celebrated GSP auction [14, 33], the ad allocation decisions are made based on the metric of bid multiplying the ad quality. The ad quality is measured by the potential actions (*e.g.*, click and purchase) of users on the displayed ads, such as click through rate (CTR) [8, 19, 42] and conversion rate (CVR) [24], which can be estimated by learning models over the rich features of users and ads and the abundant data of user-ad interaction[1]. Thus, the performance of ad allocation depends not only on the rule of auction mechanism, but also on the accuracy of learning models.

In practice, direct implementation of such a one-stage ad auction mechanism faces scalability issues, and a two-stage auction architecture is used to make a trade-off between the system scalability and ad allocation performance. The one-stage ad auction mechanism must decide the auction outcome on a set of thousands of ads within tens of milliseconds [19, 43]. However, the sophisticated learning models, such as Wide&Deep [8] and DIN [42], can not complete the CTR estimations for all the candidate ads within a limited response time. To relieve this dilemma between performance and scalability, we turn to the two-stage architecture for online ad auction, which is also adopted for large-scale online recommenders [9, 15]. An intuitive and greedy design of such a two-stage ad auction is illustrated in Figure 1a: in the first stage, called pre-auction stage, we rank the full set of N ads $\mathcal{A}_N$, and select a subset of M ads $\mathcal{A}_M$ by *bid* multiplying $\widetilde{ctr}$ from a fast and coarse ctr estimator M^c; then in the second stage, called auction stage, we determine the final ad allocation results by evaluating only on the selected ad subset via *bid* multiplying *ctr* from a sophisticated and refined ctr estimator M^r. Although this greedy two-stage auction mechanism has been widely deployed in industry [19, 35], it suffers serious loss on ad allocation performance. Due to the ability gap between the two *ctr* estimators, the coarse and the refined *ctr*s sometimes differ greatly for the same ⟨ad,user⟩ pair. Some ads with high refined *ctr* but low coarse *ctr* could be filtered out by the first stage, losing the chance of entering the second stage and winning an ad slot.

---

[1]Without loss of generality, we regard CTR as the ad quality in this work.
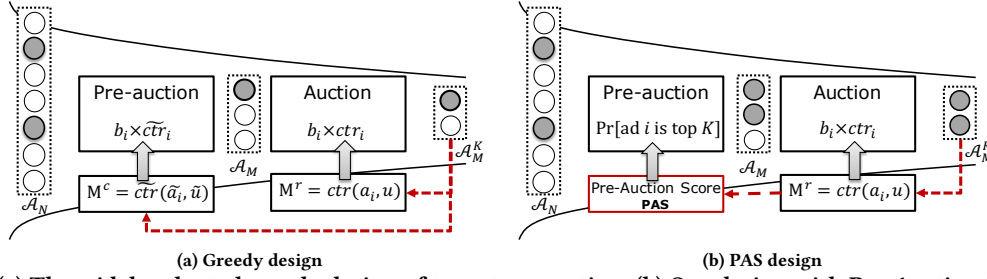
Figure 1: (a) The widely adopted greedy design of two-stage auction. (b) Our design with Pre-Auction Score (PAS).

This greedy design reflects a common misunderstanding in applying the two-stage architecture for online advertising. While optimizing ad allocation performance for the two-stage auction, just regarding each stage as a separate optimization problem, *e.g.*, conducting the individual GSP auction with the same metric of $bid \times ctr$ for ad selection in each stage, would suffer performance degradation. The auction designer should consider the interaction between the two stages (*i.e.*, the second stage will refine the estimation and selection over the subset delivered by the first stage), and figure out proper selection metrics for each stage.

In this work, we focus on the problem of designing a two-stage auction for online advertising, jointly considering the scalability of large-scale online systems and the ad allocation performance guarantee. There are two major challenges of this work. The first challenge is to characterize the conditions of the two-stage ad auction to satisfy the economic properties of *incentive compatible (IC)*, *i.e.*, the advertisers are encouraged to report their truthful value to a user click as bid, and *individual rational (IR)*, *i.e.*, the utility for an advertiser is always non-negative. The second challenge is the large decision space and then the high computational complexity of designing the two-stage auctions, *i.e.*, how can we decouple the auction design of the two stages and propose specific auctions for each stage, so that the whole two-stage auction has performance guarantee within limited response time. Although there exist many works for designing ad auction mechanisms to optimize variance of performance metrics such as social welfare and revenue [23, 33], none of them considered the two-stage auction architecture. At the same time, although the two-stage recommender system gradually attracts attention recently [9, 15, 20], the two-stage problem of online advertising differs from that of recommender systems because there are payment transfer as well as the requirement of economic properties of IC and IR in ad auctions, while neither of these two issues appears in recommender systems.

The intuition behind our proposed solution for the two-stage ad auction is as follows. First, we specify the utility model for advertisers as *value maximizers*, a proper model to capture the objectives of advertisers in online advertising [25, 36], and then obtain the characteristics of the two-stage auction to be IC and IR under this model. Then, to reduce the design space, we fix the second stage auction as the celebrated general second price (GSP) auction, and demonstrate that this decoupling preserves the optimality of performance and the properties of IC and IR for the two-stage auction. Next, we focus on the design in the first stage, *i.e.*, the pre-auction stage, and formulate it as a stochastic subset selection problem, which is to select M stochastic elements to maximize the expected sum of the realized top K elements when we regard the unrealized

*ctr*s of ads in the pre-auction stage as random variables. However, this subset selection problem is a submodular maximization with cardinality constraints, which can be proved to be NP-hard. To derive a scalable pre-auction, we turn to a closely related problem of selecting M stochastic elements which maximizes the expected recall on the realized top K elements, based on which we can define a new ad-wise selection metric, namely *Pre-Auction Score (PAS)*. We further design a learning-based implementation of this PAS metric, which can be trained with the supervision by the refined ctr estimator M^r. As PAS is a more proper metric for the pre-auction stage, our solution outperforms the widely adopted greedy design. We illustrate the detailed procedure of our two-stage auction in Figure 1b.

We summarize the main contributions of this work as follows.

- We define a new mechanism design problem of two-stage ad auction design for online advertising, which jointly considers system scalability and performance guarantee. We also demonstrate the performance degradation of a widely adopted two-stage ad auction, which ignores the interaction between the optimization problems in the two stages.
- We propose a solution for designing a two-stage ad auction for social welfare maximization. We decouple the two-stage auction as a stochastic subset selection problem in the pre-aucton stage and the GSP auction in the second stage. We propose a new ad-wise selection metric Pre-Auction Score (PAS) to solve the subset selection. The proposed two-stage ad auction still satisfies the properties of IC and IR.
- Extensive experiments on both public and industrial data demonstrate the effectiveness of our solution. On the industrial data, our solution outperforms the greedy two-stage auction by +4.35% on social welfare and +4.59% on revenue.

2 PRELIMINARIES

In this section, we describe the ad auction model, the utility model of advertisers, and the desired economic properties of ad auctions.

For an online ad platform, a page view request from a user triggers an ad auction, where a set of N advertisers $\mathcal{A}_N = \{1, \ldots, N\}$ compete for K ad slots on the page. In an ad auction, each advertiser $i \in \mathcal{A}_N$ submits a bid b_i based on her *private* value v_i, which measures the potential benefits extracted from the user click on the ad. Besides the bids of advertisers $\boldsymbol{b} = (b_1, \ldots, b_N)$, the auction mechanism also depends on the advertiser/ad features $\boldsymbol{a} = (a_1, \ldots, a_N)$ and the user features u. Here, the advertiser/ad features a_i could be ad id, category id, and etc. The user features u could be user id, click histories, and etc. The features of ads and users are applied to evaluate the quality of displaying a certain ad to a specific user. The

auction mechanism $\langle x, p \rangle$ determines the ad allocation outcomes by the allocation scheme x and the prices of the K ad slots by the payment rule p. The allocation scheme $x_i(b, a, u) = k$ represents that the ad i wins the k-th highest ad slot, or loses the auction for $k = 0$; and the payment rule $p_i(b, a, u)$ is the price that the ad i needs to pay if it is displayed and clicked.

We next describe the utility model of advertisers. The advertisers would like to maximize the advertising performance of their products, only requiring the costs to satisfy certain constraints, such as budget constraint, cost-per-click constraint and return-of-investment constraint [37, 39]. Following the industrial observations from [3, 25], *value maximizer* model [36] captures such an optimization objective of the advertisers, while the traditional *quasi-linear utility maximizer* model [30] (*i.e.*, each advertiser i maximizes $v_i - p_i$) is no longer suitable in this scenario.

Definition 2.1. (*Value Maximizer* [36]) An advertiser i is a value maximizer if she prefers the auction outcomes with a higher ad slot while keeping the payment satisfy the constraint $p_i \leq v_i$; for the same ad slot, she prefers a lower payment p_i.

In an ad auction, the advertisers might strategically misreport their values, *i.e.*, bidding $b_i \neq v_i$, to manipulate the auction outcomes for their own interests. To avoid this kind of behavior, the ad auction mechanism needs to satisfy the following economic properties:

- *Incentive Compatibility (IC)*: truthfully reporting the private value, *i.e.*, $b_i = v_i$, is the best strategy for each advertiser i.
- *Individual Rationality (IR)*: the payment of advertiser i would not exceed the reported value, *i.e.*, advertiser i pays $p_i \leq b_i$ if ad i is displayed and clicked; or pays nothing, otherwise.

With these two properties, advertisers do not need to spend efforts in computing bidding strategy, and are encouraged to participate in the auctions with no risk of deficit. The online platform also obtains the truthful and reliable advertisers' values.

For the advertisers with the utility model as a value maximizer, it has been proven in [36] that an auction mechanism satisfies IC and IR if the following two conditions are satisfied:

- **Monotonicity**: An advertiser would win the same ad slot or a higher one if she reports a higher bid.
- **Critical price**: The payment for a winning advertiser is the minimum bid that she needs to maintain the same ad slot.

The goal of the ad auction mechanism considered in this work is to maximize the expected *social welfare*, which is the sum of the expected click values of displayed ads with respective to user's stochastic click behaviors. Social welfare is a crucial metric for online advertising, as it measures the efficiency on matching advertisers and users, and is also the upper bound of the revenue which is the sum of the total payments of the ad platform.

3 PROBLEM FORMULATION

3.1 One-stage Ad Auction

One of the most widely used ad auction mechanisms is GSP auction [14, 33]. Given each advertiser i's bid b_i for a user click along with the user's ctr_i to the ad i, the GSP auction ranks all the ads with their expected click values of display, *i.e.*, $b_i \times ctr_i$, and allocates the K ad slots from the highest to the lowest following

this rank. The payment for the winning ad at the slot $k \leq K$ is $b_{(k+1)} \times ctr_{(k+1)}/b_{(k)}$, where the subscript (k) denotes the ad with the k-th highest expected click value. GSP auction is IC and IR for value-maximizing advertisers as it satisfies the conditions of monotonicity and critical price mentioned above [36]. Thus, in GSP auction we have $b_i = v_i$ for each ad i. When there is an unbiased ctr estimator[2], the GSP auction can maximize the expected social welfare [33], *i.e.*, the total expected click value of the K winning ads: $\mathsf{SumTopK}(\{v_i \times ctr_i\}_{i \in \mathcal{A}}) = \mathsf{SumTopK}(\{b_i \times ctr_i\}_{i \in \mathcal{A}})$. Here, $\mathsf{SumTopK}(\mathcal{S})$ is a set function which outputs the sum of the largest K elements in the set $\mathcal{S}$.

3.2 CTR Estimator in Ad Auction

The performance of ad auctions largely depends on the accuracy of ctr estimators. There are various kinds of machine learning models developed in the literature to estimate the ctr in different scenarios. We classify these models into two categories: the coarse but fast ctr estimator denoted by M^c and the refined but heavy ctr estimator denoted by M^r. The coarse ctr estimator $\mathsf{M}^c = \widetilde{ctr}(\tilde{a}_i, \tilde{u})$ uses light-weight learning models [19, 35], and only leverages partial ad features $\tilde{a}_i$ and partial user features $\tilde{u}$. The refined ctrs estimator $\mathsf{M}^r = ctr(a_i, u)$ can be sophisticated learning models [41, 42], and effectively leverages full ad features a_i and user features u. For example, the full user features u can be the user profiles along with a long histories of user's behaviors [41], while the partial user features $\tilde{u}$ are just some basic user profiles. The refined estimator M^r could have a complex neural network architecture, such as attention based and sequence modeling components, to produce rich user-ad cross features [38, 41, 42]. In contrast, the coarse estimator M^c might simply follow a two-tower architecture [19] or an embedding layer followed by fully connected layers [35]. With these differences, the M^r estimator consumes longer inference time but produces more accurate ctr than the M^c estimator does.

We next investigate the relation between the coarse estimator M^c and the refined estimator M^r. Suppose the models of both M^c and M^r are sufficiently trained with the same data set $\mathcal{D}$, which is independently and identically sampled from true distribution of $\langle \text{user,ad} \rangle$ pairs. For the input of the full feature (a_i, u) with the corresponding partial feature $(\tilde{a}_i, \tilde{u})$, the outputs of M^c and M^r converge as below:

$$ctr(a_i, u) = |\mathcal{D}^+(a_i, u)|/|\mathcal{D}(a_i, u)|,$$

$$\widetilde{ctr}(\tilde{a}_i, \tilde{u}) = \frac{|\mathcal{D}^+(\tilde{a}_i, \tilde{u})|}{|\mathcal{D}(\tilde{a}_i, \tilde{u})|} = \sum_{a_i|\tilde{a}_i, u|\tilde{u}} ctr(a_i, u) \frac{|\mathcal{D}(a_i, u)|}{|\mathcal{D}(\tilde{a}_i, \tilde{u})|}, \quad (1)$$

where $\mathcal{D}(*, \diamond) \subseteq \mathcal{D}$ is the subset of samples whose partial or full features are restricted to $(*, \diamond)$, and $\mathcal{D}^+$ is the subset of positive samples, *i.e.*, the clicked samples. Since $\frac{|\mathcal{D}(a_i, u)|}{|\mathcal{D}(\tilde{a}_i, \tilde{u})|} \approx \Pr[a_i, u | \tilde{a}_i, \tilde{u}]$, the relation between M^r and M^c on input (a_i, u) can be approximately expressed as

$$\begin{aligned} \widetilde{ctr}(\tilde{a}_i, \tilde{u}) &= \sum_{a_i|\tilde{a}_i, u|\tilde{u}} ctr(a_i, u) \times \Pr[a_i, u | \tilde{a}_i, \tilde{u}] \\ &= \mathbb{E}[ctr(a_i, u) | \tilde{a}_i, \tilde{u}]. \end{aligned} \quad (2)$$

According to this relation, serving the one-stage GSP auction with M^c and M^r estimators results in expected social welfare in (3a) and

---

[2]Various calibration algorithms can be applied to augment the basic ctr estimator to further reduce the bias [10].

(3b), respectively:

$$\mathbb{E}[\mathsf{SumTopK}(\{b_i \times \widetilde{ctr}(\tilde{a}_i, \tilde{u})\}_{i \in \mathcal{A}_N})] \qquad (3a)$$
$$= \mathbb{E}[\mathsf{SumTopK}(\{\mathbb{E}[b_i \times ctr(a_i, u)|\tilde{a}_i, \tilde{u}]\}_{i \in \mathcal{A}_N})]$$
$$\leq \mathbb{E}[\mathsf{SumTopK}(\{b_i \times ctr(a_i, u)\}_{i \in \mathcal{A}_N})]. \qquad (3b)$$

The inequality in (3b) is due to Jensen's inequality for convex function and the fact that SumTopK here can be regarded as a convex function over $\mathbb{R}^N$. We see that the one-stage GSP auction with the refined M^r estimator achieves strictly higher expected social welfare than that with the coarse estimator M^c.

3.3 Two-stage Ad Auction

Due to the scalability requirement of determining auction allocation and payment over thousands of ads within tens of milliseconds, we are unable to implement the one-stage GSP auction mentioned above, because applying the refined ctr estimator M^r to the total set of ads $\mathcal{A}_N$ exceeds the decision latency with limited computational resources. To make a trade-off between the optimality of social welfare and the timely response time, the architecture of two-stage ad auction is widely used in practice [19, 35]. Specifically, the first stage, called the pre-auction stage associated with an allocation scheme $\boldsymbol{x}^p$, selects a subset of $M < N$ ads, i.e., $\boldsymbol{x}^p(\mathcal{A}_N) = \mathcal{A}_M \subsetneq \mathcal{A}_N$. With the input of the selected ads $\mathcal{A}_M$ from the first stage, the second stage, called the auction stage, determines the final allocation $\boldsymbol{x}^a$ and payment $\boldsymbol{p}^a$. The pre-auction stage uses coarse but fast machine learning models with partial ad features $\tilde{a}$ and user features $\tilde{u}$ on the large set $\mathcal{A}_N$; while the second auction stage can apply more advanced and accurate ctr estimators with full ad and user features on a relatively small set $\mathcal{A}_M$. In the next section, we demonstrate that without a careful design of the two-stage auction, we may suffer from a performance degradation.

Compared with the one-stage auction design, two challenges immediately emerge for the two-stage auction design. First, how to guarantee the economic properties of IC and IR in a two-stage auction. Second, considering that the searching space for jointly designing auctions in the two stages is huge, how to decouple the design of the two-stage auction and still guarantee the ultimate auction performance. We recall that (i) GSP auction is IC and IR for value maximizer advertisers, and (ii) GSP auction can maximize the expected social welfare when there is a refined ctr estimator. Due to these two advantages of GSP auction, we can fix the second stage as GSP auction, which introduces neither the violence of IC and IR nor the loss of optimality for expected social welfare.

We focus on the auction design in the first stage, i.e., the pre-auction stage, in this work. Firstly, we coordinate the pre-auction and the GSP auction to satisfy the two conditions (monotonicity and critical price) for IC and IR properties. Since there is no payment in the pre-auction stage, we only need to guarantee that the allocation scheme $\boldsymbol{x}^p$ satisfies monotonicity, i.e., $x_i^p(\boldsymbol{b}, \tilde{a}, \tilde{u})$ is monotone increasing with respective to b_i. By doing this, we can guarantee the monotonicity property of the two-stage auction, that is when the advertiser i increases her bid, she has a higher probability to win in the pre-auction and enter the second stage, in which she obtains a not worse ad slot. Secondly, we formulate the optimization problem in the pre-auction stage. The goal of the pre-auction is to nominate a good candidate set of ads $\mathcal{A}_M$ for the second

stage GSP auction such that the expected social welfare is maximized. With the candidate ads set $\mathcal{A}_M$, the GSP auction displays the top K ads, denoted as the set $\mathcal{A}_M^K$, with the highest expected click value, $b_i \times ctr(a_i, u)$, resulting in the expected social welfare $\mathsf{SumTopK}(\{b_i \times ctr_i\}_{i \in \mathcal{A}_M}) = \sum_{i \in \mathcal{A}_M^K} b_i \times ctr_i$. In the first stage, we have no access to the refined $ctr(a_i, u)$ but only the $(\tilde{a}_i, \tilde{u})$, and thus the optimization problem for the *pre-auction* (PA) is a stochastic optimization problem:

$$\text{(PA)} \quad \max_{\boldsymbol{x}^p} \mathbb{E}_{a|\tilde{a}, u|\tilde{u}}[\mathsf{SumTopK}(\{b_i \times ctr_i(a_i, u)\}_{i \in \mathcal{A}_M})]$$
$$s.t. \quad \mathcal{A}_M = \boldsymbol{x}^p(\boldsymbol{b}, \tilde{a}, \tilde{u}), \qquad (4)$$
$$x_i^p(\boldsymbol{b}, \tilde{a}, \tilde{u}) \text{ is monotone on } b_i, \forall \boldsymbol{b}_{-i}, \tilde{a}, \tilde{u},$$

where $\boldsymbol{b}_{-i}$ is the vector of bids after removing b_i from $\boldsymbol{b} = (b_i, \boldsymbol{b}_{-i})$.

4 SUBOPTIMALITY OF GREEDY SOLUTION

In this section, we show the performance degradation of a native greedy two-stage ad auction (GDY), which is widely used in industry. The GDY auction has a simple and intuitive definition based on the refined and coarse ctr estimators $\mathsf{M}^r = ctr(a_i, u)$ and $\mathsf{M}^c = \widetilde{ctr}(\tilde{a}_i, \tilde{u})$.

Definition 4.1. In GDY, the pre-auction stage ranks all the ads $\mathcal{A}_N$ by $b_i \times \widetilde{ctr}(\tilde{a}_i, \tilde{u})$, and delivers the highest M ads $\mathcal{A}_M^g$ to the second stage, which runs a GSP auction on set $\mathcal{A}_M^g$.

We investigate whether GDY is a proper solution for the two-stage ad auction. Only when $M = K$, the pre-auction in GDY is exactly the optimal solution for the problem in (4):

$$\mathbb{E}[\mathsf{SumTopK}(\{b_i \times ctr_i\}_{i \in \mathcal{A}_M})] = \mathbb{E}[\sum_{i \in \mathcal{A}_M} b_i \times ctr_i]$$
$$= \sum_{i \in \mathcal{A}_M} \mathbb{E}[b_i \times ctr_i] = \sum_{i \in \mathcal{A}_M} b_i \times \widetilde{ctr}_i,$$

where the first equality is due to $M = K$, the second equality is due to linearity of expectation, and the third equality comes from the relation between ctr from M^r and $\widetilde{ctr}$ from M^c shown in (2). Thus, selecting the ads with the largest M values of $b_i \times \widetilde{ctr}_i$ to form $\mathcal{A}_M$ in GDY is optimal in this case. For the case of $M > K$, we use a simple example to explain the suboptimality of GDY.

Example 4.2. There are N ads with $b_1 \times \widetilde{ctr}_1 > \ldots > b_N \times \widetilde{ctr}_N$. $\forall i \leq M, ctr_i = \widetilde{ctr}_i$. An ad $j > M$ has two possible CTR realizations, $ctr_j = t \times \widetilde{ctr}_j$ with probability $1/t$, and $ctr_j = \epsilon$ with probability $1 - 1/t$, where $t > 0$ is a large number such that $b_j \times t \times \widetilde{ctr}_j > b_K \times \widetilde{ctr}_K = b_K \times ctr_K$. GDY first selects the top M ads (without the ad j) in the pre-auction stage, and then displays the top K ads to the user in the second stage. The resulted expected social welfare is $\sum_{i=1}^{K} b_i \times ctr_i$. However, if the pre-auction stage selects $\mathcal{A}_M = \{1, \ldots, M-1, j\}$, then after the second stage GSP auction, the expected social welfare outperforms that of GDY:

$$\mathbb{E}[\mathsf{SumTopK}(\{b_i \times ctr_i\}_{i \in \mathcal{A}_M})]$$
$$= \frac{1}{t}(\sum_{i=1}^{K-1} b_i \times ctr_i + b_j \times t \times \widetilde{ctr}_j) + (1 - \frac{1}{t})\sum_{i=1}^{K} b_i \times ctr_i$$
$$> \sum_{i=1}^{K} b_i \times ctr_i.$$

The greedy two-stage auction would encounter the scenarios similar to Example 4.2, resulting in performance degradation in practice. Some ad (like ad j in the example) gets a low coarse $\widetilde{ctr}$ from M^c but its refined ctr from M^r is high. If the pre-auction stage is aware that the second stage GSP auction will further refine the ctr estimation, the pre-auction stage can make a better decision, selecting some ads with potentially high refined ctr to achieve a

higher social welfare. This example uses the relation between the coarse estimator M^c and refined estimator M^r shown in (2), *i.e.*, the pre-auction stage can regard the unrealized refined $ctr(a_i, u)$ as a random variable from a distribution with the mean value $\widetilde{ctr}(\tilde{a}_i, \tilde{u})$. This problem is a subset selection over a set of random variables in the literature, and the theoretical analysis have been considered in team selection problem [22] and other general background [29].

The main insight we want to deliver here is that the widely deployed greedy solution, regarding the design in each of the two stages as social welfare maximization separately and using the same selection metric ($b \times \widetilde{ctr}$ or $b \times ctr$) in both stages, would instead suffer a suboptimal social welfare. When we design a two-stage auction, we should consider the interaction between the two stages, *i.e.*, the second stage will refine the estimation and conduct the ad allocation over the subset delivered by the first stage, and design proper selection metrics for each stage, to guarantee the overall ad performance of the two-stage ad auction.

5 PRE-AUCTION DESIGN

We first show the computational complexity of solving the pre-auction problem defined in (4). To reduce the complexity, we then propose an ad-wise metric called pre-auction score (PAS) for scalable ad selection in the pre-auction stage. We also design a learning based implementation for PAS in practice. The detailed proofs in this section can be found in Appendix B.

5.1 Complexity of Pre-auction Problem

We first prove that, even we relax the constraint of monotonicity on the allocation x_i^p of the problem PA in (4), the resulting *simplified version of PA* (SimPA) is still intractable.

$$(\text{SimPA}) \quad \max_{\mathcal{A}_M \subseteq \mathcal{A}_N} \quad \mathbb{E}_{ctr}[\text{SumTopK}(\{b_i \times ctr_i\}_{i \in \mathcal{A}_M})]$$
$$s.t. \quad |\mathcal{A}_M| \leq M,$$

where ctr_i is a random variable.

PROPOSITION 5.1. *The SimPA problem is NP-hard.*

In fact, SimPA is a submodular maximization with a cardinality constraint. For a ground set $\mathcal{A}$, a set function $g : 2^{\mathcal{A}} \to \mathbb{R}$ is submodular if $\forall \mathcal{S} \subseteq \mathcal{T} \subset \mathcal{A}$ and $\forall j \in \mathcal{A} \backslash \mathcal{T}$, $g(\mathcal{S} \cup \{j\}) - g(\mathcal{S}) \geq g(\mathcal{T} \cup \{j\}) - g(\mathcal{T})$.

PROPOSITION 5.2. *In SimPA, the objective $\mathbb{E}_{ctr}[\text{SumTopK}(\{b_i \times ctr_i\}_{i \in \mathcal{S}})]$ is a submodular set function with respective to the set $\mathcal{S}$.*

There is no apparent tractable and scalable solution for this submodular optimization in the setting of pre-auction stage: (i) The brute force algorithm, which evaluates the selection $\mathcal{A}_M$ at set-wise scale, causes an intractable $O(\binom{N}{M})$ computation. (ii) The classical approximation algorithms [4, 31] select ads sequentially based on their marginal contributions, failing to run in an ad-wise parallel way, and thus are still not scalable in practice. (iii) We emphasize that from the perspective of the pre-auction stage, ctr as well as its explicit distribution are unknown, which introduces difficulty on even evaluating the objective submodular function. Thus, even the parallel approximation algorithms [1, 2, 6], which rely on submodular function evaluation, are not suitable here.

Considering the scalability in online service of pre-auction stage, we propose an ad-wise metric for subset selection in Section 5.2.

With this ad-wise metric, we can evaluate each ad in a parallel way, and avoid evaluation of the submodular set function. To tackle the lack of explicit distribution of ctr, we propose learning based implementation for the ad-wise metric in Section 5.3.

5.2 Pre-auction Score

We design a tractable and scalable ad-wise metric for subset selection in the pre-auction stage, which supports parallel evaluation on each ad. Since that social welfare maximization of pre-auction is an NP-hard problem, it is hopeless to directly obtain an ad-wise metric as the exact solution for social welfare maximization. The key insight here is that we can turn to an objective closely related to social welfare maximization, and then derive a corresponding ad-wise metric. The closely related objective, called *recall maximization* (PA-R), is to select the set of ads $\mathcal{A}_M$ in the pre-auction stage to cover as much final top K ads of $\mathcal{A}_N$ in the auction stage,

$$(\text{PA-R}) \quad \max_{\mathcal{A}_M \subseteq \mathcal{A}_N} \mathbb{E}_{ctr}[\text{SumTopK}(\{\mathbb{1}[i \in \mathcal{A}_N^K]\}_{i \in \mathcal{A}_M})], \quad (5)$$

where the final top K ads of $\mathcal{A}_N$, denoted as $\mathcal{A}_N^K$, are the ads with the highest K $b_i \times ctr(a_i, u)$ from $\mathcal{A}_N$ in the auction stage. If $\mathcal{A}_M$ covers all the top K ads $\mathcal{A}_N^K$, it achieves the maximum social welfare of the one-stage auction, *i.e.*, the upper bound social welfare of two-stage ad auction.

We now derive the exact ad-wise metric, such that the pre-auction can rank and retrieve the top M ads $\mathcal{A}_M$ according to this metric to optimize PA-R. Consider any fixed subset $\mathcal{A}_M$, the expected recall on $\mathcal{A}_N^K$ is

$$\begin{aligned} & \mathbb{E}_{ctr}[\text{SumTopK}(\{\mathbb{1}[i \in \mathcal{A}_N^K]\}_{i \in \mathcal{A}_M})] \\ = \ & \sum_{i \in \mathcal{A}_M} \mathbb{E}_{ctr}[\mathbb{1}[i \in \mathcal{A}_N^K]] \\ = \ & \sum_{i \in \mathcal{A}_M} \Pr_{ctr}[i \in \mathcal{A}_N^K], \end{aligned} \quad (6)$$

where the first equality is due to the linearity of expectation and the second equality is due to the definition of indicator function. The above equation shows that selecting the ads with the M largest $\Pr_{ctr}[i \in \mathcal{A}_N^K]$, *i.e.*, the probabilities of being in $\mathcal{A}_N^K$, maximizes the expected recall on $\mathcal{A}_N^K$. Hence, we obtain an ad-wise metric for subset selection in the pre-auction, denoted as f_i for each ad i:

$$(\text{PAS}) \quad f_i(\boldsymbol{b}, \tilde{\boldsymbol{a}}, \tilde{u}) = \Pr_{ctr}[i \in \mathcal{A}_N^K]. \quad (7)$$

We call this metric as *Pre-Auction Score (PAS)*. Note that although PAS is an ad-wise metric, it is still allowed to use the information of the whole ad set $\mathcal{A}_N$ in the pre-auction stage, *i.e.*, $\boldsymbol{b}, \tilde{\boldsymbol{a}}, \tilde{u}$, to facilitate the calculation of the probability. According to the definition of set $\mathcal{A}_N^K$ and PAS in (7), for any advertiser i, partial ads features $\tilde{\boldsymbol{a}}$, user features $\tilde{u}$ and other advertiser's bid b_{-i}, the advertiser i's PAS is monotonely increasing with respective to b_i. Thus, if the pre-auction ranks all the ads $\mathcal{A}_N$ with the metric PAS and selects the top M ads, it satisfies the monotonicity of allocation in the pre-auction, and hence satisfies the IC of the two-stage auction.

5.3 Learning Based Pre-auction Score

Computing the metric PAS requires an explicit form of distribution $\Pr[ctr|\boldsymbol{b}, \tilde{\boldsymbol{a}}, \tilde{u}]$, which might be complicated within the unknown and interdependent online environment. To overcome this difficulty, we use parametrized neural networks to implement a learning based PAS f_i^θ, such that the permutation of ranking by f_i^θ can

approximate the permutation of ranking by the original PAS f_i in (7). We use supervised learning to determine the parameters θ of the neural networks f_i^{θ}. For each training sample, the features are $(\boldsymbol{b}, \tilde{\boldsymbol{a}}, \tilde{u})$, i.e., the N bids, the partial ad features and the partial user features; and the label is the N-dim vector $\boldsymbol{y}$, where the element is $y_i = b_i \times ctr(a_i, u)$ for each ad i. During the training process, we need $ctr(a_i, u)$ from the refined estimator M^r for all the ads in $\mathcal{A}_N$ to compute the label of a sample. But we only obtain $ctr(a_i, u)$ for ad $i \in \mathcal{A}_M$ during the online service, because only the ads $\mathcal{A}_M$ enter the second auction stage and are evaluated by the refined estimator. Thus, we need an offline refined estimator M^r to produce $ctr(a_i, u)$ for the ads in $\mathcal{A}_N \backslash \mathcal{A}_M$.

Follow the Plackett-Luce probability model, which is widely used in learning the distribution of permutations [5, 17], we assume that the permutation π of ranking by $y_i = b_i \times ctr(a_i, u)$ is a sample from the distribution of permutations which is defined with N parameters $\{y_i\}_{i \in \mathcal{A}}$ as follows

$$\Pr[\pi|\boldsymbol{y}] = \prod_{i=1}^{N} \frac{y_{\pi(i)}}{\sum_{k=i}^{N} y_{\pi(k)}}, \quad (8)$$

where $\pi(k)$ is the ad with rank k. We can easily verified that this definition of probability satisfies two desired properties: (i) It is normalized, i.e., $\sum_{\pi} \Pr[\pi|\boldsymbol{y}] = 1$; (ii) The top 1 probability is:

$$\Pr[i \text{ is the top 1} |\boldsymbol{y}] = \frac{y_i}{\sum_{k=1}^{N} y_k}. \quad (9)$$

LEMMA 5.3. For $y_i > y_j$ and $K > 1$, $\Pr[i \in \mathcal{A}_N^K] > \Pr[j \in \mathcal{A}_N^K]$.

With Lemma.5.3, we can use the rank by y_i to represent the rank by PAS $\Pr[i \in \mathcal{A}_N^K]$. Let the neural networks directly output f_i^{θ} as logits for approximate y_i. The top-1 probability of the ad i determined by logits $\{f_i^{\theta}\}_{i \in \mathcal{A}_N}$ is calculated as

$$\Pr[i \text{ is the top 1} |\boldsymbol{f}^{\theta}(\boldsymbol{b}, \tilde{\boldsymbol{a}}, \tilde{u})] = \frac{\exp(f_i^{\theta})}{\sum_{k \in \mathcal{A}_N} \exp(f_k^{\theta})}.$$

Given the sample set $\mathcal{D}_f$, we minimize the cross entropy between the sample distribution and the parametric distribution by $\boldsymbol{f}^{\theta}$, so the loss function is

$$L = -\frac{1}{|\mathcal{D}_f|} \sum_{j \in \mathcal{D}_f} \sum_{i=1}^{N} \Pr[i \text{ is the top 1} |\boldsymbol{y}^j] \\ \times \log \Pr[i \text{ is the top 1} |\boldsymbol{f}^{\theta}(\boldsymbol{b}^j, \tilde{\boldsymbol{a}}^j, \tilde{u}^j)] \quad (10)$$

where the superscript j means the j-th sample in $\mathcal{D}_f$.

6 EXPERIMENTS

We evaluate the effectiveness of our proposed two-stage auction solution on both public and industrial datasets.

6.1 Settings for Public Dataset

The public dataset Amazon Dataset [27] contains product reviews and metadata from Amazon [18, 28]. We conduct experiments on the subset called Books, which contains 603K user reviews, 367K items and 1600 categories. We regard reviews as user clicks and regard items as ads. The full features of a user u include a user_id,

and a list of user's reviewed items along with categories in history, i.e., $u = \langle \text{user\_id, hist\_items\_id, hist\_cate\_id} \rangle$. The length of each user's item list is at least 5. The full ad features a_i is $a_i = \langle \text{item\_id, cate\_id} \rangle$. We define the partial user feature $\tilde{u}$ as $\tilde{u} = \langle \text{user\_id, hist\_items\_id[:3], hist\_cate\_id[:3]} \rangle$, i.e., only the first three items in user's history are used in $\tilde{u}$. The partial ad features $\tilde{a}_i$ is defined as the same as the full a_i. Based on this dataset, we simulate the process of two-stage auction and generate 40,000 auctions. In each auction, there are 1000 randomly selected ads. The bid of each ad is independently sampled from the uniform distribution.

To simulate the second stage GSP auction, we use ad and user's full features $\langle a_i, u \rangle$ to train Deep Interested Networks (DIN), a baseline ctr estimator [42], as the refined estimator M^r to generate $ctr(a_i, u)$, and the label $b_i \times ctr(a_i, u)$ for each ad. To simulate GDY for comparison, we use ad and user's partial features $\langle \tilde{a}_i, \tilde{u} \rangle$ to train a model with embedding followed by fully connected layers (FCN) [35], as the coarse and fast estimator M^c to generate $\widetilde{ctr}(\tilde{a}_i, \tilde{u})$ for each ad. The training samples for M^r DIN and M^c FCN are the same, i.e., the same $\langle \text{ad, user} \rangle$ pairs from the dataset as positive samples, along with the same 1:1 negative sampling. From [19], the resulting $ctr = \frac{p}{p+(1-p)/\eta}$, where p is the output of the trained estimator and η is the negative down sampling rate. The difference between ctr and $\widetilde{ctr}$ gets larger as η decreases, since the difference between raw output p from M^r and M^c is approximately amplified by a factor $1/\eta$. We consider two settings, Public-1 and Public-5, whose η are 0.01 and 0.05, respectively. Public-1 simulates the ctr in real-world online advertising. We would like to verify whether our solution can still outperform GDY in the scenario simulated by Public-5, where the difference between ctr and $\widetilde{ctr}$ is smaller and the GDY is able to perform relatively better.

6.2 Settings for Industrial Dataset

The industrial dataset comes from the log of a two-stage ad auction in a leading e-commerce platform Taobao, running GDY defined as Definition 4.1. We randomly sample 30K auction records from the logged data on January 8th 2021. There are 700 ads in each auction instance. For each ad, the features are: (i) $\widetilde{ctr}(\tilde{a}_i, \tilde{u})$ from M^c, some historical statistics such as historical averaged refined ctr and cvr of this ad. We regard these estimated values as cross features of the ad and user's partial features $\langle \tilde{a}_i, \tilde{u} \rangle$; (ii) Ad information like bid b_i, category and selling price of the product in the ad. The estimated $ctr(a_i, u)$ from M^r are used to generate the label $b_i \times ctr(a_i, u)$. Note that under our assumption of value maximizing advertisers, GDY satisfies IC and IR. Therefore, we can regard the logged bids as advertiser's truthful values for user clicks. Then, we can use the logged bids to compute social welfare and revenue during simulating other IC and IR auction mechanisms.

6.3 Evaluation Metrics

We consider the following common used metrics for ad auction evaluation. We recall that $\mathcal{A}_N^K$ is the true top K ads in the whole set $\mathcal{A}_N$ while $\mathcal{A}_M^K$ is the top K ads in the selected subset $\mathcal{A}_M$.

- Social welfare rate: $SWr@K = \frac{\sum_{i \in \mathcal{A}_M^K} b_i \times ctr_i}{\sum_{j \in \mathcal{A}_N^K} b_j \times ctr_j}$.
- Top K Recall: $Recall@K = \sum_{i \in \mathcal{A}_M^K} \mathbb{1}[i \in \mathcal{A}_N^K]/K$.

Table 1: Results of different methods on data setting Public-1. $N = 1000, M = 50$. **Notations in a table cell: average ± standard deviation (Improvement over GDY) in 20 runs.**

	Recall@1	Recall@5	Recall@10	SWr@5	REVr@5
GDY	0.8383 ± 0.0022, (0%)	0.7378 ± 0.0024, (0%)	0.6597 ± 0.0029, (0%)	0.9252 ± 0.0007, (0%)	0.9028 ± 0.0010, (0%)
REGCTR	0.8440 ± 0.0058, (+0.57%)	0.7469 ± 0.0079, (+0.91%)	0.6698 ± 0.0079, (+1.01%)	0.9234 ± 0.0048, (-0.18%)	0.9026 ± 0.0050, (-0.02%)
REG	0.8506 ± 0.0068, (+1.23%)	0.7483 ± 0.0109, (+1.05%)	0.6709 ± 0.0112, (+1.12%)	0.9274 ± 0.0041, (+0.22%)	0.9058 ± 0.0050, (+0.30%)
PAS	**0.9093 ± 0.0021, (+7.10%)**	**0.8639 ± 0.0045, (+12.61%)**	**0.8246 ± 0.0057, (+16.49%)**	**0.9613 ± 0.0011, (+3.61%)**	**0.9519 ± 0.0015, (+4.91%)**

Table 2: Results of different methods on data setting Public-5. $N = 1000, M = 50$. **Notations are the same as Table.1.**

	Recall@1	Recall@5	Recall@10	SWr@5	REVr@5
GDY	0.8359 ± 0.0023, (0%)	0.7390 ± 0.0026, (0%)	0.6623 ± 0.0025, (0%)	0.9389 ± 0.0006, (0%)	0.9213 ± 0.0008, (0%)
REGCTR	0.9073 ± 0.0029, (+7.14%)	0.8502 ± 0.0028, (+11.12%)	0.7972 ± 0.0032, (+13.49%)	0.9669 ± 0.0008, (+2.80%)	0.9578 ± 0.0010, (+3.65%)
REG	0.9094 ± 0.0037, (+7.35%)	0.8521 ± 0.0051, (+11.31%)	0.7982 ± 0.0057, (+13.59%)	0.9673 ± 0.0016, (+2.84%)	0.9583 ± 0.0018, (+3.70%)
PAS	**0.9155 ± 0.0071, (+7.96%)**	**0.8745 ± 0.0075, (+13.55%)**	**0.8363 ± 0.0080, (+17.40%)**	**0.9720 ± 0.0022, (+3.31%)**	**0.9651 ± 0.0027, (+4.38%)**

Table 3: Results of different methods on data setting Industrial. $N = 700, M = 30$. **Notations are the same as Table.1.**

	Recall@1	Recall@5	Recall@10	SWr@5	REVr@5
GDY	0.4745 ± 0.0046, (0%)	0.3740 ± 0.0019, (0%)	0.3175 ± 0.0013, (0%)	0.7808 ± 0.0013, (0%)	0.7483 ± 0.0010, (0%)
REGCTR	0.5156 ± 0.0047, (+4.11%)	0.4043 ± 0.0028, (+3.03%)	0.3451 ± 0.0024, (+2.76%)	0.8081 ± 0.0017, (+2.73%)	0.7764 ± 0.0022, (+2.81%)
REG	0.5118 ± 0.0059, (+3.73%)	0.4048 ± 0.0033, (+3.08%)	0.3474 ± 0.0027, (+2.99%)	0.8089 ± 0.0022, (+2.81%)	0.7783 ± 0.0025, (+3.00%)
PAS	**0.5351 ± 0.0035, (+6.06%)**	**0.4226 ± 0.0024, (+4.86%)**	**0.3635 ± 0.0017, (+4.60%)**	**0.8243 ± 0.0013, (+4.35%)**	**0.7942 ± 0.0007, (+4.59%)**

- Revenue rate: $REVr@K = REV(\mathcal{A}_M^K)/REV(\mathcal{A}_N^K)$, where revenue REV is under GSP auction and the subscript (k) means the advertiser with the k-th highest $b \times ctr$ in the corresponding set, and $REV(\mathcal{A}) = \sum_{k=1}^{K} b_{(k+1)} \times ctr_{(k+1)}$.

6.4 Methods for Comparison

To prove the effectiveness of our proposed solution with PAS metric and its learning based implementation, we introduce the following baselines of pre-auction for comparison. Due to the practical deployment requirements, we only focus on the methods which select ads using *ad-wise metric* in the pre-auction stage. Implementations of these ad-wise metrics are also restricted to use the same partial features $\langle \tilde{a}, u \rangle$ as PAS uses.

- Greedy (GDY): As described in Definition 4.1, the rank score of GDY for ad i is simply $b_i \times \widetilde{ctr}_i$, where $\widetilde{ctr}_i$ is the output of the coarse estimator M^c.
- Regression to ctr_i (REGCTR): We use ctr_i as label, and use ad and user's partial features $\langle \tilde{a}_i, \tilde{u} \rangle$ to train a regression model with mean square loss. The rank score is bid times output of the regression model.
- Regression to $b_i \times ctr_i$ (REG): We use $b_i \times ctr_i$ as label, and use the partial ad, user features and the bid, *i.e.*, $b_i, \tilde{a}_i, \tilde{u}$, to train a regression model with mean square loss.

The neural network structure and input for REGCTR, REG, and PAS are almost the same, except that REGCTR network lacks the input of bid. The reason we introduce REGCTR and REG for comparison is to demonstrate that PAS is a more proper selection metric for pre-auction. In each repeated experiment, we split data into training, validation, and test set with 3:1:1, and apply early stopping with metric $SWr@5$ on validation set for all three methods.

6.5 Performance Comparison

Results of different methods on data setting Public-1, Public-5 and Industrial are given in Table 1-3. The tables show average metrics, standard deviation and the improvement over GDY in 20 runs with 20 distinctive random seeds for each data setting and methods.

We can obtain the following observations from Table 1-3. (i) We can see that our proposed PAS outperforms all baseline methods in each data setting and on each metric. For instance, PAS improves $SWr@5$ by +3.61%, +3.31% and +4.35% comparing with the widely used GDY in Public-1, Public-5 and Industrial, respectively. (ii) As REGCTR, REG and PAS have better performance than GDY on most data settings and metrics, we conclude that pre-auction stage can benefit from the supervision by the second stage's information. (iii) REGCTR and REG fall behind our proposed PAS. The reason can be that while REGCTR and REG are enforced to learn a regression on ctr_i or $b_i \times ctr_i$, PAS which models the probability of being top K, is a more proper selection metric for the pre-auction stage.

Next, we compare the performance results in Table 1-2 for Public-1 and Public-5, the only difference between which is the down negative sampling rate η. The smaller the η is, the larger the gap between coarse $\widetilde{ctr}$ and refined ctr is. This indicates that GDY on Public-1 has a worse performance than on Public-5, which can also be observed from the evaluation results. We can also see that REGCTR and REG achieve much worse performance on Public-1 than on Public-5. For example, $SWr@5$ of REG on Public-5 is 0.967 which outperforms $SWr@5$ of GDY by +2.84%, but the corresponding values turn out to be only 0.927 and +0.22% on Public-1. However, PAS is more stable than REG and REGCTR, and achieves similar good results on both Public-1 and Public-5.

In Figure 2, an auction instance for each of the three data settings is illustrated to show the results of the four methods working in a two-stage auction. The figures for Public-1 and Public-5 are from the same auction instance, while only the down negative sampling rates for calculating ctr are different. We first explain the three large figures in the above of Figure 2 which illustrate the results of method GDY. The x-axis is the rank by $bid \times ctr$ with the refined ctr estimator, and here we show the top 200 ads. A blue and red point with the same x coordinate are associated with the same ad. The y-axis for blue points is the normalized value of $bid \times ctr$, while the y-axis for red points is the normalized value of $bid \times \widetilde{ctr}$. The black horizontal line shows the threshold of the normalized value $bid \times \widetilde{ctr}$ for entering the second auction stage in GDY. Red points above the black line enter the second auction stage, and
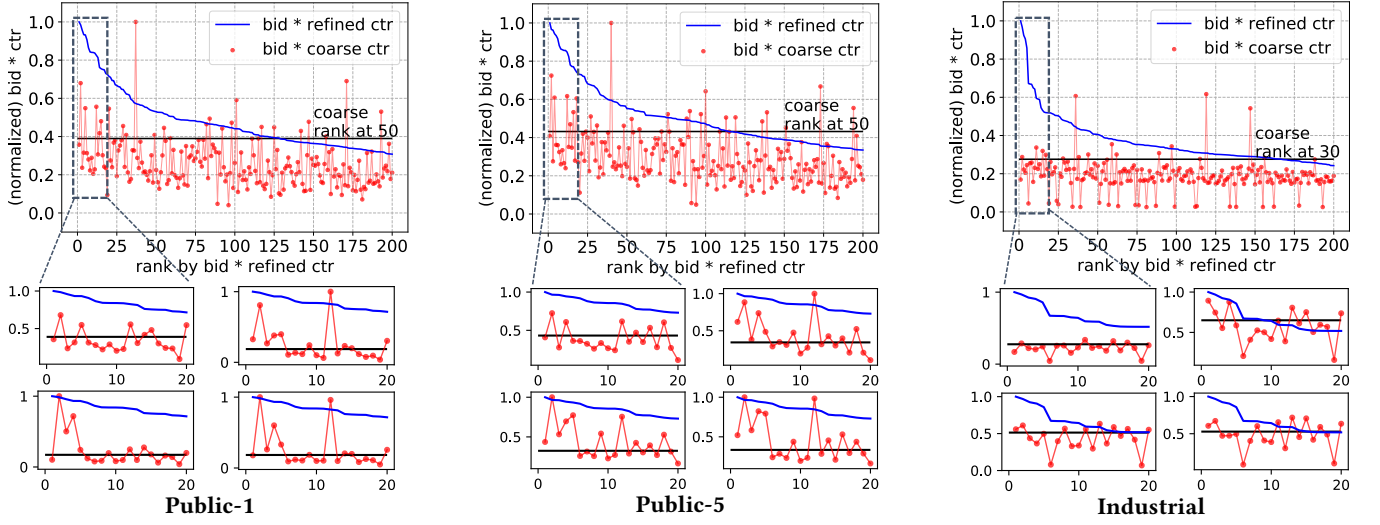
Figure 2: Illustration of two-stage auction with methods GDY, PAS, REGCTR and REG on data Public-1, Public-5 and Industrial. Layout of the four small figures: top-left, GDY; top-right, PAS; bottom-left, REGCTR; bottom-right, REG.

Table 4: Failure rates of perturbation tests, with unit $\times 10^{-5}$ (average ± standard deviation in 20 runs).

	GDY	REGCTR	REG	PAS
Public-1	0	0	10.4 ± 5.27	0 ± 0
Public-5	0	0	0 ± 0	0 ± 0
Industrial	0	0	4.63 ± 5.69	1.40 ± 2.34

the most left 5 red points among them obtain the top 5 ranks in the second auction stage and win ad slots. Next, we explain the small figures which show the top 20 ads in four different methods. Similarly, y-axis for the red points is the normalized score of pre-auction in the method, while the black horizontal line is the corresponding threshold for entering the second stage. We can see that variant methods result in different rankings in the pre-auction and different black horizontal lines. For example in the third column for Industrial data, the ad slots are allocated to the ads with the rank of $bid \times ctr$ as $\{2, 11, 15, 17, 20\}$ (GDY), $\{1, 2, 4, 11, 13\}$ (PAS), $\{1, 2, 8, 11, 13\}$ (REGCTR), and $\{1, 2, 8, 11, 13\}$ (REG), and thus PAS have a better $Recall@5$ and $SWr@5$ than GDY, REGCTR, and REG in this example.

6.6 IC Testing

As we have mentioned in Section 2 that in order to guarantee the economic properties of IC and IR for the auction, we require that the allocation for each advertiser is monotonely increasing with respective to her bid. That is to say, the pre-auction stage need to satisfy: for any advertiser i, given partial features for all ads $\tilde{a}$ and for the user $\tilde{u}$, as well as bids from other advertisers b_{-i}, there exists a threshold b_i^t that the ad i enters the second stage auction if and only if $b_i \geq b_i^t$. To show that learning based PAS can achieve the monotonicity approximately, we conduct counterfactual perturbation on each advertiser's bid in the logged data and evaluate the violation of monotonicity condition. This is a common method for IC testing of ad auction mechanisms [11, 12].

Specifically, we sample 1000 auctions from the test set, and conduct IC test for each ad in each auction. For an ad i, all its features to PAS model remain the same except that we replace b_i with $\alpha \times bid_i$, where $\alpha \in S_p = \{0.2 \times j | j = 1, \ldots, 10\}$ is a multiplicative perturbation factor from interval $[0, 2]$. All the features of other ads in this auction remain the same. We simulate the two-stage auction on all 10 perturbation factors to check whether the PAS model passes the perturbation tests, *i.e.*, (i) $\exists \underline{\alpha} \in S_p$ such that ad i can enter the auction stage with $\alpha \times b_i \ \forall \alpha \geq \underline{\alpha}$; or (ii) ad i can not enter auction stage with $\alpha \times b_i$ for any $\alpha \in S_p$.

We conduct 1000×1000 tests, for 1000 auctions and 1000 advertisers per auction, in each of Public-1 and Public-5, and 1000×700 in Industrial data. Table 4 shows the average failure rates and the standard deviations of 20 runs. The low failure rates show that even we apply no deliberate design on PAS model for a guarantee of strict monotonicity, the learning based PAS learns an approximate monotonicity automatically. An intuitive reason might be that there is an inherent pattern in the supervised data that bid has a positive effect on the objective of model. We also test REG and obtain similarly low failure rates. For GDY and REGCTR, their metrics are monotone with respective to bid, so their allocations of pre-auction are naturally monotone.

7 CONCLUSION

We have studied a novel problem of designing a large-scale two-stage ad auction, which consists of a pre-auction stage and an auction stage, to maximize social welfare for online advertising. We illustrated social welfare loss of a widely adopted design due to its improper selection metric in the pre-auction stage, which ignores the relation between the two auction stages. We have designed a practical two-stage ad auction solution which guarantees IC and IR for value maximizers, where the second stage is GSP auction and the pre-auction stage selects the ad subset with an ad-wise metric called Pre-Auction Score (PAS). We have further proposed a

learning based implementation of PAS. Experiment results on both public and industrial dataset have shown that our proposed solution outperforms the greedy design significantly on social welfare and revenue of the two-stage ad auction.

REFERENCES

[1] Eric Balkanski, Aviad Rubinstein, and Yaron Singer. 2019. An exponential speedup in parallel running time for submodular maximization without loss in approximation. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 283–302.

[2] Eric Balkanski and Yaron Singer. 2018. The adaptive complexity of maximizing a submodular function. In *Proceedings of the 50th annual ACM SIGACT symposium on theory of computing*. 1138–1151.

[3] Santiago Balseiro, Yuan Deng, Jieming Mao, Vahab Mirrokni, and Song Zuo. 2021. The Landscape of Auto-Bidding Auctions: Value Versus Utility Maximization. *Available at SSRN 3785579* (2021).

[4] Niv Buchbinder, Moran Feldman, Joseph Naor, and Roy Schwartz. 2014. Submodular maximization with cardinality constraints. In *Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms*. 1433–1452.

[5] Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. 2007. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th International Conference on Machine Learning*. 129–136.

[6] Chandra Chekuri and Kent Quanrud. 2019. Submodular function maximization in parallel via the multilinear relaxation. In *Proceedings of the 30th Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 303–322.

[7] Minmin Chen, Alex Beutel, Paul Covington, Sagar Jain, Francois Belletti, and Ed H Chi. 2019. Top-k off-policy correction for a REINFORCE recommender system. In *Proceedings of the 12th ACM International Conference on Web Search and Data Mining*. 456–464.

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

[9] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems*. 191–198.

[10] Chao Deng, Hao Wang, Qing Tan, Jian Xu, and Kun Gai. 2020. Calibrating user response predictions in online advertising. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. 208–223.

[11] Yuan Deng and Sébastien Lahaie. 2019. Testing dynamic incentive compatibility in display ad auctions. In *Proceedings of the 25th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 1616–1624.

[12] Yuan Deng, Sébastien Lahaie, Vahab Mirrokni, and Song Zuo. 2020. A data-driven metric of incentive compatibility. In *Proceedings of the Web Conference 2020*. 1796–1806.

[13] Paul Dütting, Zhe Feng, Harikrishna Narasimhan, David Parkes, and Sai Srivatsa Ravindranath. 2019. Optimal Auctions through Deep Learning. In *Proceedings of the 36th International Conference on Machine Learning*. 1706–1715.

[14] Benjamin Edelman, Michael Ostrovsky, and Michael Schwarz. 2007. Internet advertising and the generalized second-price auction: Selling billions of dollars worth of keywords. *American Economic Review* 97, 1 (2007), 242–259.

[15] Chantat Eksombatchai, Pranav Jindal, Jerry Zitao Liu, Yuchen Liu, Rahul Sharma, Charles Sugnet, Mark Ulrich, and Jure Leskovec. 2018. Pixie: A system for recommending 3+ billion items to 200+ million users in real-time. In *Proceedings of the Web Conference 2018*. 1775–1784.

[16] Negin Golrezaei, Max Lin, Vahab Mirrokni, and Hamid Nazerzadeh. 2021. Boosted Second Price Auctions: Revenue Optimization for Heterogeneous Bidders. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 447–457.

[17] John Guiver and Edward Snelson. 2009. Bayesian inference for Plackett-Luce ranking models. In *Proceedings of the 26th International Conference on Machine Learning*. 377–384.

[18] Ruining He and Julian McAuley. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *Proceedings of the Web Conference 2016*. 507–517.

[19] Xinran He, Junfeng Pan, Ou Jin, Tianbing Xu, Bo Liu, Tao Xu, Yanxin Shi, Antoine Atallah, Ralf Herbrich, Stuart Bowers, et al. 2014. Practical lessons from predicting clicks on ads at facebook. In *Proceedings of the 8th International Workshop on Data Mining for Online Advertising*. 1–9.

[20] Jiri Hron, Karl Krauth, Michael I Jordan, and Niki Kilbertus. 2021. On component interactions in two-stage recommender systems. *arXiv preprint arXiv:2106.14979* (2021).

[21] Wang-Cheng Kang and Julian McAuley. 2019. Candidate generation with binary codes for large-scale top-n recommendation. In *Proceedings of the 28th ACM international conference on information and knowledge management*. 1523–1532.

[22] Jon Kleinberg and Maithra Raghu. 2018. Team performance with test scores. *ACM Transactions on Economics and Computation (TEAC)* 6, 3-4 (2018), 1–26.

[23] Sébastien Lahaie and David M Pennock. 2007. Revenue analysis of a family of ranking rules for keyword auctions. In *Proceedings of the 8th ACM Conference on Electronic Commerce*. 50–56.

[24] Kuang-chih Lee, Burkay Orten, Ali Dasdan, and Wentong Li. 2012. Estimating conversion rate in display advertising from past erformance data. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 768–776.

[25] Xiangyu Liu, Chuan Yu, Zhilin Zhang, Zhenzhe Zheng, Yu Rong, Hongtao Lv, Da Huo, Yiqing Wang, Dagui Chen, Jian Xu, Fan Wu, Guihai Chen, and Xiaoqiang Zhu. 2021. Neural auction: End-to-end learning of auction mechanisms for e-commerce advertising. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery &; Data Mining*. 3354–3364.

[26] Jiaqi Ma, Zhe Zhao, Xinyang Yi, Ji Yang, Minmin Chen, Jiaxi Tang, Lichan Hong, and Ed H Chi. 2020. Off-policy learning in two-stage recommender systems. In *Proceedings of the Web Conference 2020*. 463–473.

[27] Julian McAuley. 2015. Amazon Product Data. http://jmcauley.ucsd.edu/data/amazon/.

[28] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. 2015. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR Conference on Research and Development in Information Retrieval*. 43–52.

[29] Aranyak Mehta, Uri Nadav, Alexandros Psomas, and Aviad Rubinstein. 2020. Hitting the high notes: Subset selection for maximizing expected order statistics. *Advances in Neural Information Processing Systems* 33 (2020).

[30] Roger B Myerson. 1981. Optimal auction design. *Mathematics of Operations Research* 6, 1 (1981), 58–73.

[31] George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. 1978. An analysis of approximations for maximizing submodular set functions—I. *Mathematical Programming* 14, 1 (1978), 265–294.

[32] Weiran Shen, Binghui Peng, Hanpeng Liu, Michael Zhang, Ruohan Qian, Yan Hong, Zhi Guo, Zongyao Ding, Pengjun Lu, and Pingzhong Tang. 2020. Reinforcement mechanism design: With applications to dynamic pricing in sponsored search auctions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 2236–2243.

[33] Hal R Varian. 2007. Position auctions. *international Journal of industrial Organization* 25, 6 (2007), 1163–1178.

[34] Hal R Varian and Christopher Harris. 2014. The VCG auction in theory and practice. *American Economic Review* 104, 5 (2014), 442–45.

[35] Zhe Wang, Liqin Zhao, Biye Jiang, Guorui Zhou, Xiaoqiang Zhu, and Kun Gai. 2020. COLD: Towards the Next Generation of Pre-Ranking System. *arXiv preprint arXiv:2007.16122* (2020).

[36] Christopher A Wilkens, Ruggiero Cavallo, and Rad Niazadeh. 2017. GSP: the cinderella of mechanism design. In *Proceedings of the Web Conference 2017*. 25–32.

[37] Di Wu, Xiujun Chen, Xun Yang, Hao Wang, Qing Tan, Xiaoxun Zhang, Jian Xu, and Kun Gai. 2018. Budget constrained bidding by model-free reinforcement learning in display advertising. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. 1443–1451.

[38] Jun Xiao, Hao Ye, Xiangnan He, Hanwang Zhang, Fei Wu, and Tat-Seng Chua. 2017. Attentional factorization machines: learning the weight of feature interactions via attention networks. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*. 3119–3125.

[39] Xun Yang, Yasong Li, Hao Wang, Di Wu, Qing Tan, Jian Xu, and Kun Gai. 2019. Bid optimization by multivariable control in display advertising. In *Proceedings of the 25th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 1966–1974.

[40] Xinyang Yi, Ji Yang, Lichan Hong, Derek Zhiyuan Cheng, Lukasz Heldt, Aditee Kumthekar, Zhe Zhao, Li Wei, and Ed Chi. 2019. Sampling-bias-corrected neural modeling for large corpus item recommendations. In *Proceedings of the 13th ACM Conference on Recommender Systems*. 269–277.

[41] Guorui Zhou, Na Mou, Ying Fan, Qi Pi, Weijie Bian, Chang Zhou, Xiaoqiang Zhu, and Kun Gai. 2019. Deep interest evolution network for click-through rate prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 5941–5948.

[42] Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. 2018. Deep interest network for click-through rate prediction. In *Proceedings of the 24th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 1059–1068.

[43] Han Zhu, Junqi Jin, Chang Tan, Fei Pan, Yifan Zeng, Han Li, and Kun Gai. 2017. Optimized cost per click in taobao display advertising. In *Proceedings of the 23rd ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 2191–2200.

A RELATED WORK

Online advertising auction. In the market of online ad auction, traditional auction mechanisms like general second price auction (GSP) [14, 33] and Vickrey-Clark-Groove auction (VCG) [34] are widely used. Recently, many parametric mechanisms learning from data are proposed to optimize performance metrics of the auction market, *e.g.* squashed GSP [23], boosted second price auction [16], and dynamic reserve price via reinforcement learning [32] for the task of revenue maximization. The paradigm of differentiable mechanism design via deep learning proposed by Dütting *et al.* [13] is also applied to multi-objective optimization in online ad auctions [25].

Two-stage recommender. Two-stage architectures with candidate generation followed by ranking have been widely adopted in large scale industrial recommenders. Despite their popularity, the literature on two-stage recommenders is relatively scarce. To improve computation efficiency [21, 40] and recommendation quality [7] are still the two most important problems of recommenders even in two-stage setting. Interaction between the two stages and its influence on system performance are studied in [7, 20, 26]. Our problem of two-stage auction system differs from the two-stage recommenders. Because of the payment transfer and requirement of economic properties, an auction system needs quantifiable and explainable evaluation of user's interests like *ctr*, while a recommender only determines the order of items.

B PROOFS

PROOF FOR PROPOSITION 5.1. We prove that SimPA is NP-hard even when $K = 1$ by reducing a well-known NP-hard problem set cover to SimPA. We first define the general set cover problem. The universal set is $\mathcal{U} = \{u_1, \ldots, u_L\}$. There is a collection $\mathcal{T} = \{\mathcal{S}_1, \ldots, \mathcal{S}_N\}$ where $\forall i \leq N, \mathcal{S}_i \subset \mathcal{U}$ is a subset of universal set $\mathcal{U}$. The set cover problem is to decide whether there is a collection $\mathcal{T}' \subset \mathcal{T}, |\mathcal{T}'| \leq M$ such that $\bigcup_{\mathcal{S}_i \in \mathcal{T}'} \mathcal{S}_i = \mathcal{U}$. Next, for any instance of the set cover problem, we define the corresponding SimPA instance. Randomly sample a non-empty subset $\mathcal{U}'$ from universal set $\mathcal{U}$, then we define a set of random variables $\boldsymbol{ctr} = \{ctr_1, \ldots, ctr_N\}$ where each $ctr_i \in \{0, \frac{1}{b_i}\}$ is corresponding to the above set $\mathcal{S}_i$: if any element in $\mathcal{S}_i$ are sampled in $\mathcal{U}'$, $ctr_i = \frac{1}{b_i}$ and $b_i \times ctr_i = 1$; otherwise, $ctr_i = 0$ and $b_i \times ctr_i = 0$. Therefore, there exists a collection $\mathcal{T}' \subset \mathcal{T}, |\mathcal{T}'| \leq M$ if and only if there exists a set $\mathcal{A}_M$ that $i \in \mathcal{A}_M \Leftrightarrow \mathcal{S}_i \in \mathcal{T}'$, such that $\mathbb{E}_{\boldsymbol{ctr}}[\text{SumTopK}(\{b_i \times ctr_i\}_{\mathcal{A}_M})] = 1$ for $K = 1$. □

PROOF FOR PROPOSITION 5.2. We can easily verify that SumTopK is a submodular set function. For any fixed value of $\boldsymbol{ctr}$, and $\forall \mathcal{S} \subseteq \mathcal{T} \subset \mathcal{A}_N$ and $j \in \mathcal{A}_N \backslash \mathcal{T}$,

$$\text{SumTopK}(\{b_i \times ctr_i\}_{i \in \mathcal{S} \cup \{j\}}) - \text{SumTopK}(\{b_i \times ctr_i\}_{i \in \mathcal{S}})$$
$$\geq \quad \text{SumTopK}(\{b_i \times ctr_i\}_{i \in \mathcal{T} \cup \{j\}}) - \text{SumTopK}(\{b_i \times ctr_i\}_{i \in \mathcal{T}}).$$

Then, we take expectation over the distribution of $\boldsymbol{ctr}$ on both sides of the above inequality,

$$\mathbb{E}_{\boldsymbol{ctr}}[\text{SumTopK}(\{b_i \times ctr_i\}_{i \in \mathcal{S} \cup \{j\}})]$$
$$-\mathbb{E}_{\boldsymbol{ctr}}[\text{SumTopK}(\{b_i \times ctr_i\}_{i \in \mathcal{S}})]$$
$$\geq \quad \mathbb{E}_{\boldsymbol{ctr}}[\text{SumTopK}(\{b_i \times ctr_i\}_{i \in \mathcal{T} \cup \{j\}})]$$
$$-\mathbb{E}_{\boldsymbol{ctr}}[\text{SumTopK}(\{b_i \times ctr_i\}_{i \in \mathcal{T}})],$$

and finish the proof. □

PROOF FOR LEMMA 5.3. For any pair of permutations (π, π') that satisfy (1) $\pi(i) = \pi'(j) \leq K$; (2) $\pi(j) = \pi'(i) > K$; and (3) $\pi(k) = \pi'(k)$ for all $k \neq i \wedge k \neq j$, we can easily verify that $\Pr[\pi|\boldsymbol{y}] > \Pr[\pi'|\boldsymbol{y}]$. Then, summing up on all these pairs of (π, π'), we have

$$\Pr[i \in \mathcal{A}_N^K \wedge j \notin \mathcal{A}_N^K] > \Pr[j \in \mathcal{A}_N^K \wedge i \notin \mathcal{A}_N^K],$$

and then

$$\Pr[i \in \mathcal{A}_N^K] = \Pr[i, j \in \mathcal{A}_N^K] + \Pr[i \in \mathcal{A}_N^K \wedge j \notin \mathcal{A}_N^K]$$
$$\geq \Pr[i, j \in \mathcal{A}_N^K] + \Pr[j \in \mathcal{A}_N^K \wedge i \notin \mathcal{A}_N^K] = \Pr[j \in \mathcal{A}_N^K].$$

□