Decision Support Systems 56 (2013) 379–386

Contents lists available at ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss

Online learning for auction mechanism in bandit setting

Di He [a], Wei Chen [b], Liwei Wang [a,*], Tie-Yan Liu [b]

[a] Key Laboratory of Machine Perception, MOE, School of Electronics Engineering and Computer Science, Peking University, 100871 Beijing, PR China
[b] Microsoft Research Asia, Beijing, PR China

ARTICLE INFO

Article history:
Received 7 November 2012
Received in revised form 11 June 2013
Accepted 12 July 2013
Available online 26 July 2013

Keywords:
Armed bandit problem
Mechanism design
Online advertising

ABSTRACT

This paper is concerned with online learning of the optimal auction mechanism for sponsored search in a bandit setting. Previous works take the click-through rates of ads to be fixed and known to the search engine and use this information to design optimal auction mechanism. However, the assumption is not practical since ads can only receive clicks when they are shown to users. To tackle this problem, we propose to use online learning for auction mechanism design. To be specific, this task corresponds to a new type of bandit problem, which we call the armed bandit problem with shared information (AB-SI). In the AB-SI problem, the arm space (corresponding to the parameter space of the auction mechanism which can be discrete or continuous) is partitioned into a finite number of clusters (corresponding to the finite number of rankings of the ads), and the arms in the same cluster share the explored information (i.e., the click-through rates of the ads in the same ranked list) when any arm from the cluster is pulled. We propose two upper-confidence-bound algorithms called UCB-SI1 and UCB-SI2 to tackle this new problem in discrete-armed bandit and continuum-armed bandit setting respectively. We show that when the total number of arms is finite, the regret bound obtained by UCB-SI1 algorithm is tighter than the classical UCB1 algorithm. In the continuum-armed bandit setting, our proposed UCB-SI2 algorithm can handle a larger classes of reward function and achieve a regret bound of $O(T^{2/3}(d\ln T)^{1/3})$, where d is the pseudo dimension for the real-valued reward function class. Experimental results show that the proposed algorithms can significantly outperform several classical online learning methods on synthetic data.

1. Introduction

Nowadays, online advertising has become one of the most profitable business models for internet companies. Sponsored search, as a major type of online advertising, is a revenue powerhouse for search engines. Keyword auction is the central mechanism in sponsored search, which determines the ads to be present to the users (which we call ranking) and the per-click prices to charge the corresponding advertisers (which we call pricing).

The keyword auction mechanism works as follows. When a web user submits a query to search engine, the search engine not only delivers organic search results to him/her, but also shows real-time sponsored search results, i.e., advertisements (see Fig. 1). As a dominant industry practice, the search engine will charge an advertiser only when a user clicks on his/her ad. This is referred to as cost-per-click pricing rule (CPC). Generalized Second Price Auction (GSP) [1,2] is a widely used mechanism for CPC, in which the ads are ranked according to a function of the ad quality and its bid price, and the per-click price of a displayed ad equals the minimal bid price for the owner of the ad to maintain the current rank position. Different ways of computing quality score have been used in the literature, for example, Yahoo! once used a constant quality score in early 2000s [1], and Google uses the predicted click-through rate nowadays [2].

With different quality score functions, different ads will be shown to the users and advertisers will receive different charged prices if their ads are clicked. Thus the quality score function will highly affect the search engine's performance, e.g., revenue. In the literature, there are several works on revenue maximization for the search engine, from either machine learning or game theory perspective. In [3,4], the authors assume that the click-through rate and bidding price of each ad are known and fixed, and then propose a machine learning approach to find the revenue-optimal quality score function based on historical log data. In [5,6], the authors assume that the advertisers have full information, and the click through-rate of the ads are known to the search engines, then different quality score functions can be compared with respect to the worst-case revenue in symmetric Nash equilibrium or Bayesian Nash equilibrium (Fig. 2).

However, in most of the previous works, a key assumption is that the click-through rates of all ads are known to the search engine and never changed. This is seldom true in practice. In real applications, there are usually hundreds of advertisers bidding on one keyword, and only a small number of ads can be shown on the search result page and receive clicks. If one ad has never been shown to the users, the probability of the ad being clicked cannot be observed; Furthermore, even if one ad has been shown to users in history, the probability of it being clicked is difficult to estimate because the variance is large if the number of

* Corresponding author. Tel.: +86 10 6275 6657; fax: +86 10 6275 5569.
E-mail address: wanglw@cis.pku.edu.cn (L. Wang).

0167-9236/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.dss.2013.07.004

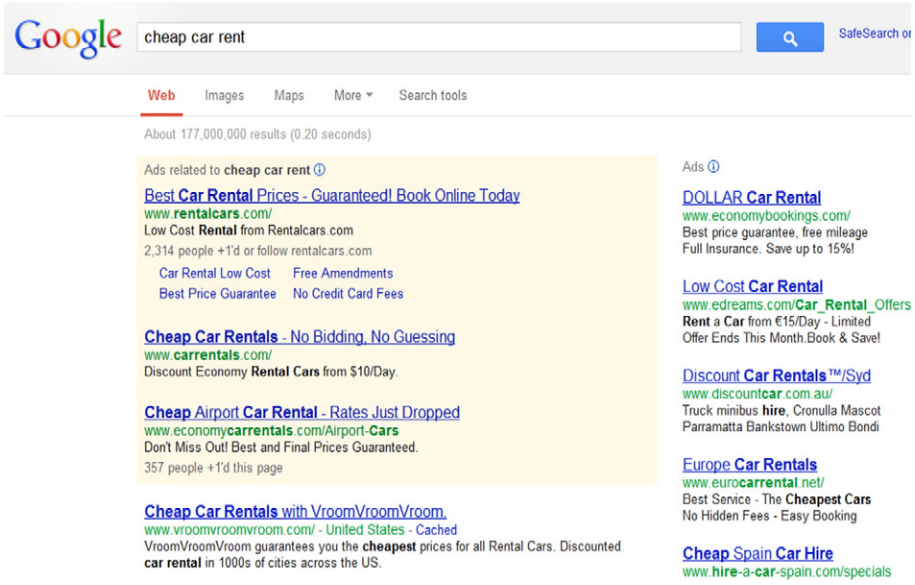

Fig. 1. The displayed ads for query "cheap car rent".

observations is small. Such limitations make the pervious machine learning methods or game theoretic analysis for revenue maximization not practical.

To tackle this problem, we propose online exploring the users' click behaviors as well as using the explored information for auction mechanism design. In particular, we propose to user the bandit algorithms, where the quality score function in GSP mechanism corresponds to the arm, and the performance (e.g., search engine revenue) corresponds to the reward function.

Our setting has several new features. First, in our task, while the quality score function (arm) may come from a continuous function class, only the top-ranked ads will have impact on the revenue (since only these top-ranked ads will be shown to the users). This makes the reward function discontinue. Furthermore, the reward function is non-convex due to the complex per-click pricing scheme. This type of reward function has never been studied in literature of bandit problems. Second, the arms in our new bandit problem share the explored information. No matter how different the quality score functions are, as long as the ranked list of ads produced is the same, users will give the same kinds of feedback (because users can only see the ranked list of ads, but not the scores for these ads). Therefore, those arms that produce the same ranking result have high dependency on their rewards, and the explored information about user clicks can be shared among them.

Considering the aforementioned uniqueness, we propose a new type of armed bandit problem, called the armed bandit problem with shared information (AB-SI). Specifically, the AB-SI problem has a two-layer structure. In the first layer, there are $K(K < \infty)$ clusters, with arms putting into different clusters according to their dependencies. In the second layer, within a cluster, the number of arms may be finite or infinite, when exploring any arm in a given cluster, the obtained information can be shared with other arms in the cluster.

To handle the aforementioned problem, we propose a stepwise online–offline learning algorithm, which we call the UCB-SI algorithm in general (which can be regarded as a generalization of the standard UCB algorithm [7]). In the online learning phase of the algorithm, one of the clusters is selected according to the best empirical performance of the arms in the cluster as well as a confidence value, then the best

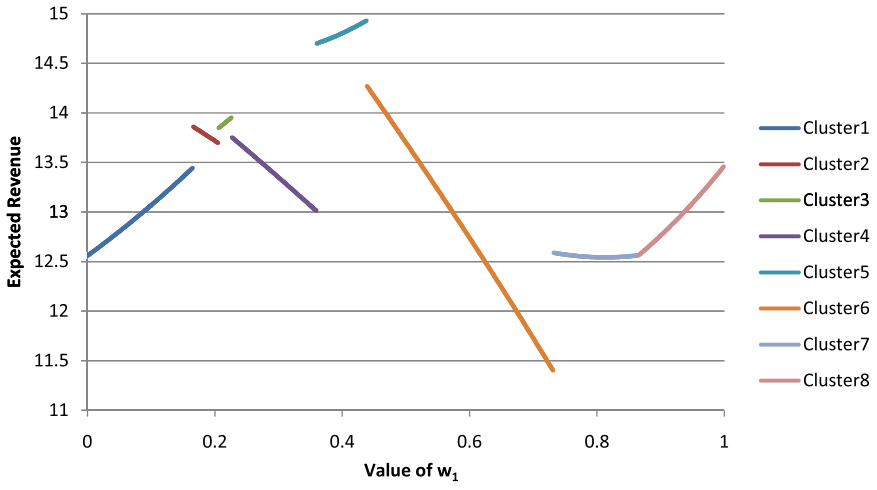

Fig. 2. The dependency between expected revenue and arms.

arm in the selected cluster will be pulled and new explored information for the cluster is received. In the offline learning phase, the arm with the best empirical performance in a cluster is updated by using supervised learning algorithms based on all explored information. These two phases alternate and one can eventually find the optimal arm (which corresponds to the optimal auction mechanism).

We analyzed the theoretical properties (e.g., the regret bound) of the two proposed UCB-SI algorithms, UCB-SI1 for multi-armed bandit problem, and UCB-SI2 for continuum-armed bandit problem, both in an i.i.d information setting. We show that when the total number of arms is finite, the proposed UCB-SI1 algorithm can achieve a tighter regret bound than the classical UCB algorithm; when the arm space is continuous, UCB-SI2 algorithm also has a reasonable regret bound of $O(T^{2/3}(d\ln T)^{1/3})$, where d is the pseudo dimension for the real-valued reward function class.

1.1. Related work

Revenue maximization is an important aspect of mechanism design, and many people have studied the optimization of search engine revenue [3–6,8]. These studies can be categorized into two groups. The first group tackles the task from a machine learning perspective. In [3,4], the authors propose to simultaneously optimize the revenue and relevance of the auction mechanism on historical bidding and click-through data. These works usually assume that the bidding prices are fixed and the click-through rates of ads have been fully explored in the historical logs. The second group addresses the problem from a game-theoretic perspective. In [5] the worst-case revenue in the symmetric Nash equilibrium is maximized, and in [8,6], the Bayesian optimal auction mechanism design is investigated with the value distribution of the bidders as public knowledge. In these works, one usually assumes that the values/bids (or their distributions) of the advertisers, the click-through rates and the auction mechanism of the search engine are accessible as public knowledge. However, in reality, the number of ads is huge and only the click probability of the ads that have been shown before can be estimated. As a result, the practical values of the aforementioned attempts are not very clear.

Our approach of using online learning methods is similar to several online learning algorithms in the literature [7,9–15]. In [7,9,10], the armed bandit problem with a finite number of arms is introduced and classical UCB1 and Exp3 algorithm are developed. In [13–15], the authors deal with continuum-armed bandit problem, under different smooth assumptions. In [13], the authors prove the first sub-linear regret bound for Lipschitz-continuous reward function in 1-dimensional arm space, and in [14,15] the authors develop a nearly-tight regret bound in the same setting and give a sub-linear regret bound for Lipschitz-continuous reward functions in general d-dimensional arm space. However, in our problem, the arm space is continuous while the reward function is discontinuous and non-convex due to the ranking and pricing rules. Therefore, our problem setting is more complex than the conventional armed bandit problems. Our dependency assumption among arms is similar to the dependent-arm setting in [12]. Both works assume that there are clusters in the arm space. However, there are also clear differences between them. In [12], it is assumed that the reward of the arms in each cluster comes from a generative model and this is where the dependency results from. In contrast, in our model, the dependency comes from the shared information within a cluster.

Another branch of our related works is to apply online learning to mechanism design in sponsored search. The general idea is to collect information about the click-through rates of ads to design better mechanism in the future. In [16], the authors firstly formulated ad placement with budgets constraints as a multi-armed bandit problem and design optimal first-price auction mechanism; Truthful multi-armed bandit mechanisms are developed using online schemes in [17,11], for example, in [17], the authors develop an exploration–exploitation separated online scheme and prove a regret bound with order $\Omega(T^{2/3})$.

2. Motivations

As mentioned in the introduction, keyword auction is one of the central mechanisms in sponsored search. Assume for keyword q, there are M advertisements and the bid prices are $\mathbf{b} = \{b_1, \cdots, b_M\}$. When this keyword is asked by a web user, the search engine will run an auction to decide which ads should be shown to the user and the per-click price to be charged from the corresponding advertisers.

Several auction mechanisms have been developed for sponsored search, and the GSP [1] family is among the most popular ones. In GSP mechanism, both the ranking and pricing of ads are determined by a scoring function $s : ad \times bid \to R$. The scoring function could depend on many factors, such as the bid price, and the quality of ad. Commonly, a set of features $x_i \in \mathcal{X}$ is extracted for ad i, and the score function is defined as a product of quality score function f and the bid, i.e., $s(x_i, b_i) = f(x_i) \times b_i$. Assume there are L slots to display ads, with scores $\{s_1, \cdots, s_M\}$, the expected revenue for the keyword is written as follows,

$$\sum_{r=1}^{L} CTR(\pi_s(r)) p_s(r),$$

where $\pi_s(r)$ is the index of ad ranked at position r by the scoring function s, $CTR(i)$ is ad i's click-through rate, and $p_s(r) = \frac{s_{\pi_s(r+1)} b_{\pi_s(r+1)}}{s_{\pi_s(r)}}$ is the charged price for the ad at rth slot.

With different quality score functions, the GSP mechanisms produce various ranking lists and charged prices. Denote $\mathcal{F}$ as the quality score function space. The goal of auction mechanism design is to find $f^* \in \mathcal{F}$ that can maximize the expected revenue defined as above.

However, the click-through rates of the ads are unknown to the search engine, because users can only give clicks to the ads that are presented to them. If an ad has never been shown to users in any ranking lists, the search engine will have no information from the users about its click probability. This suggests that online exploration is essential to get these information. In the next section, we propose a general armed bandit setting that motivated from this learning problem.

3. An online learning framework for auction mechanism design

In this section, we introduce an online learning framework to characterize the auction mechanism design problem, which is called armed bandit problem with shared information (AB-SI), and then propose a general online learning algorithm for AB-SI.

3.1. Armed bandit problem with shared information

In this subsection, we use bandit algorithms to explore necessary information for the optimal auction mechanism design. In this bandit setting, the quality score function corresponds to the arm, and the performance (e.g., search engine revenue) corresponds to the reward function.

Given advertisers' bids $\mathbf{b}$ fixed,[1] users' click behaviors are the only information that we need to explore. For ease of discussion, we number all possible top-L ranking lists of ads generated by the quality score function class $\mathcal{F}$ as 1,..., K. When the keyword is issued at time t, the search engine will rank the ads according to the bidding prices and quality score function f, and put the top-L ad list on the search result page. Then the user will make clicks on these ads (as feedback to the sponsored search system). Denote the user click behavior as Y_k^t if the kth ad list is shown to him/her, in which Y_k^t is a binary vector indicating

[1] In this paper, we simply assume that the advertisers will not change their bids, there are many empirical evidences about such stable states, e.g. [18].

which ad the user clicks. Thus the revenue of search engine at time t can be considered as a function of quality score function f and Y_k^t.

For different quality score functions, as long as the produced top-L ranking list is the same, the user will see no difference and will provide the same kinds of click information. That is, the expected revenue of the quality score functions that produce the same top-L rankings are highly dependent. This is very different from the assumptions in conventional multi-armed bandit problem where the rewards of arms are independent of each other. We call this new setting armed bandit with shared information (AB-SI), which is abstracted as below.

Denote $\mathcal{F}$ as the arm space, which can be a convex and compact space or of finite size. Denote $\mathcal{F}_1, ..., \mathcal{F}_K$ as a set of pre-defined K clusters, where the clusters are disjoint subsets of $\mathcal{F}$ and $\mathcal{F} = \cup_k \mathcal{F}_k$. Denote $\sigma(f)$ as the index of cluster that contains arm f. At each round t, the environment draws an information vector $Y^t = (Y_1^t, ..., Y_K^t)$, where Y_k^t is independently sampled from a distribution $P_k(y)$ for any k. We assume the reward function has the following form

$$R\left(f, Y^t\right) = r_{\sigma(f)}\left(f, Y^t_{\sigma(f)}\right)$$
$$Y_k^t \sim P_k(y), k = 1, 2, ..., K.$$

We use expected regret to measure the performance of the online learning algorithm. Define the expected reward $R(f) = E_{Y^t} R\left(f, Y^t\right)$, then the regret can be written as below.

$$Regret = \sum_{t=1}^{T}\left[max_{f\in\mathcal{F}} R(f) - R\left(f_{Alg}^t\right)\right] \quad (1)$$

where f_{Alg}^t is the arm that the online algorithm pulls at round t.

3.2. The UCB-SI algorithm

In this subsection, we provide an algorithm to tackle the AB-SI problem, which is called UCB-SI algorithm (see Algorithm 1). This algorithm is a stepwise online–offline learning algorithm. In the online phase, one of the clusters is selected according to the best empirical performance of the arms in each cluster as well as a confidence value which depends on how many times the cluster has been explored before, then the arm with the best empirical performance from the selected cluster is pulled and new information for the cluster is received. In the offline phase, the arm with the best empirical performance in each cluster is updated by using any supervised learning algorithms based on all explored information within the cluster. These two phases alternate and we can eventually find the optimal arm.

More details of the UCB-SI algorithm are explained as follows. Denote $L_k(t)$ as the number of times that UCB-SI algorithm pulls arms from cluster $\mathcal{F}_k$ before round t. In the first K rounds, the algorithm explores information for each cluster once,[2] and receives the information Y_k^k for each cluster k. At any round $t > K$, based on explored information in each cluster $\mathcal{F}_k$, the algorithm employs supervised learning algorithm to find the empirical optimal arm in $\mathcal{F}_k$, denoted as $\hat{f}_{k,t}$.

$$\hat{f}_{k,t} = argmax_{f\in F_k}\hat{R}(f, L_k(t)) = argmax_{f\in F_k}\frac{1}{L_k(t)}\sum_{j=1}^{L_k(t)} r_k\left(f, Y_k^{t_{k,j}}\right) \quad (2)$$

where $t_{k,j}$ is the round number that the algorithm pulls arms from cluster $\mathcal{F}_k$ for the jth time.

[2] The algorithm begins with a K-round exploration to initialize the user's click probabilities. After the initialization, each ad in each ad list will receive an impression and a click/non-click signal, then the supervised learning in the offline phase can be worked.

We give a score to each cluster k, which is the sum of the best empirical performance among the arms in $\mathcal{F}_k$, and a confidence value $\eta_{k,t}$ which depends on the number of times that cluster $\mathcal{F}_k$ is selected. And the cluster with the largest score will be selected and the arm with the best empirical performance in the cluster will be pulled. The details about how to set the confidence value $\eta_{k,t}$ will be discussed in the next section.

Algorithm 1 *UCB-SI*

Require: $\mathcal{F}_1, \mathcal{F}_2, \cdots, \mathcal{F}_K$, where $\cup\mathcal{F}_k = \mathcal{F}$.

1: Randomly select inner points f_k in F_k, $\forall k \in \{1, ..., K\}$.
2: Let $L_k(t)$ records the times that the algorithm pulls arms from cluster $\mathcal{F}_k$ before round t.
3: Pull arm $f_1, ..., f_K$ in the first K rounds.
4: **for** $t = K+1, \cdots, T$ **do**
5: For each cluster $\mathcal{F}_k$, call any supervised learning algorithms to find empirical optimal arm $\hat{f}_{k,t} = argmax_{f\in F_k}\hat{R}(f, L_k(t)) = argmax_{f\in F_k}\frac{1}{L_k(t)}\sum_{j=1}^{L_k(t)} r_k(f, Y_k^{t_{k,j}})$.
6: Select cluster $\mathcal{F}_{k_t}$ w.r.t the largest value of $\hat{R}(\hat{f}_{k,t}, L_k(t)) + \eta_{k,t}$.
7: Pull arm $\hat{f}_{k_t,t}$ at time t, and receive $Y_{k_t}^t$.
8: **end for**

4. Regret analysis

In this section, we discuss the regret bounds for two UCB-SI algorithms by setting different $\eta_{k,t}$. In particular, we will make the discussions in two settings. The first setting assumes that the size of arm space is finite while the second works for the arm space of infinite size. In the first setting, we develop UCB-SI1 algorithm and show that its regret bound is sharper than that of the classical UCB1 algorithm. In the second setting, we design UCB-SI2 algorithm which manages to obtain a regret bound of $O(T^{2/3}(d\ln T)^{1/3})$ for a general class of reward function, where d is the pseudo dimension for the real-valued reward function class.

4.1. Arm space of finite size

In this subsection, we develop UCB-SI1 algorithm for multi-armed bandit problem and prove a sub-linear regret bound. According to Algorithm 1, for this case in the offline part, we can just select arms by their empirical performance instead of supervised learning. For simplicity and without loss of generality, we assume every cluster has N arms in it, and denote $f_{k,n}$ as the nth arm in cluster $\mathcal{F}_k$. When we use the classical multi-armed bandit algorithms, e.g. UBC1 algorithm, we will have KN arms in total and get a regret bound of $O(KN\ln T)$. In the following, we propose UCB-SI1 algorithm in which $\eta_{k,t}$ is set to be $\sqrt{\frac{2\ln t + \frac{1}{2}\ln N}{L_k(t)}}$, we show that the algorithm can achieve a sharper regret bound: $O(K\ln T + N\ln T)$.

Theorem 1. (Regret Bound for UCB-SI1 Algorithm) By setting $\eta_{k,t}$ to be $\sqrt{\frac{2\ln t + \frac{1}{2}\ln N}{L_k(t)}}$, the expected regret of UCB-SI1 algorithm is $O(K\ln T + N\ln T)$.

The proof of the theorem is based on the following two lemmas. The following notations are used in both of the lemmas and the proof of the theorem. Denote f_k^* as the arm that produces the largest expected reward in cluster $\mathcal{F}_k$ and let k_0 be the cluster containing the arm that produces the largest expected reward in $\mathcal{F}$. We call a cluster $\mathcal{F}_k$ sub-optimal, if $k \neq k_0$, and call an arm f sub-sub-optimal, if f is in a sub-optimal cluster $\mathcal{F}_k$ and $f \neq f_k^*$.

With these notations, the lemmas basically indicate that the expected number of times that UCB-SI1 algorithm selects a sub-optimal cluster and the number of times that the algorithm pulls a sub-sub-optimal arm are at most $O(\ln T)$ and $O(\ln\ln T)$ separately.

Lemma 1. *Denote* $\Delta_k = R\left(f^*_{k_0}\right) - R(f^*_k)$. *Before any round T, the expected number of times that the UCB-SI1 algorithm pulls arms from the sub-optimal cluster* $\mathcal{F}_k$ *is no more than* $\frac{8\ln T}{\Delta_k^2} + \frac{2\ln N}{\Delta_k^2} + 1 + \frac{\pi^2}{3}$.

Proof. The proof follows the proof of Theorem 1 in [7] and we give a proof sketch here. For a sub-optimal cluster $\mathcal{F}_k$, it is easy to show that for any positive integer l,

$$L_k(T) \le l + \sum_{t=1}^{\infty}\sum_{s=1}^{t=1}\sum_{s_k=l}^{t-1} \left\{\widehat{R}\left(\widehat{f}_{k_0,s}, s\right) + \sqrt{\frac{2\ln t + \frac{1}{2}\ln N}{s}} \le \widehat{R}\left(\widehat{f}_{k,s_k}, s_k\right) + \sqrt{\frac{2\ln t + \frac{1}{2}\ln N}{s_k}}\right\}. \tag{3}$$

Since $R\left(f^*_{k_0}\right) - \widehat{R}\left(\widehat{f}_{F_{k_0},s}, s\right) \le R\left(f^*_{k_0}\right) - \widehat{R}\left(f^*_{k_0}, s\right)$ and $\widehat{R}\left(\widehat{f}_{F_k,s_k}, s_k\right) - R(f^*_k) \le \widehat{R}\left(\widehat{f}_{k,s_k}, s_k\right) - R\left(\widehat{f}_{k,s_k}\right)$. By letting $l = \frac{8\ln T + 2\ln N}{\Delta_k^2}$, when $s_k > l$, the indicator function in the RHS of (3) equals 1 only if at least one of the following inequalities holds,

$$R\left(f^*_{k_0}\right) - \widehat{R}\left(f^*_{k_0}, s\right) \ge \sqrt{\frac{2\ln t + \frac{1}{2}\ln N}{s}} \tag{4}$$

$$\widehat{R}\left(\widehat{f}_{k,s_k}, s_k\right) - R\left(\widehat{f}_{k,s_k}\right) \ge \sqrt{\frac{2\ln t + \frac{1}{2}\ln N}{s_k}}. \tag{5}$$

Applying Chernoff bound in (4) and generalization error bound in (5), we have

$$P\left(R\left(f^*_{k_0}\right) - \widehat{R}\left(f^*_{k_0}, s\right) \ge \sqrt{\frac{2\ln t + \frac{1}{2}\ln N}{s}}\right) \le e^{-2s\frac{2\ln t + \frac{1}{2}\ln N}{s}} \le t^{-4} \tag{6}$$

$$P\left(\widehat{R}\left(\widehat{f}_{k,s_k}, s_k\right) - R\left(\widehat{f}_{k,s_k}\right) \ge \sqrt{\frac{2\ln t + \frac{1}{2}\ln N}{s_k}}\right) \le N e^{-2s_k\frac{2\ln t + \frac{1}{2}\ln N}{s_k}} \le t^{-4}. \tag{7}$$

Combining (6), (7) and taking expectation on both sides of (3), we can prove the theorem. □

Lemma 2. *Denote* $L_{k,n}(T)$ *as the number of times that UCB-SI1 algorithm pulls the nth arm in cluster* $\mathcal{F}_k$ *before round T. If* $f_{k,n} \neq f^*_k$, $EL_{k,n}(T)$ *is no more than* $E\left[\frac{8\ln L_k(T)}{\Delta_{k,n}^2} + 1 + \frac{\pi^2}{3}\right]$, *where* $\Delta_{k,n} = R(f^*_k) - R(f_{k,n})$.

Since all information is shared within cluster k, we can consider the supervised learning here as a specific UCB1 pulling policy with a same confidence value for each arm. Thus the proof is similar to that of Lemma 1 and we prove Theorem 1 as follows.

Proof of Theorem 1. We prove the theorem by means of error decomposition, The decomposition divides the regret into two parts, the sufferings from picking a sub-optimal cluster and the sufferings from picking a sub-optimal arm in any clusters.

$$\begin{aligned} Regret &= E\left(\sum_{t=1}^{T}\left[R\left(f^*_{k_0}\right) - R\left(f^t_{Alg}\right)\right]\right) = E\left(\sum_{k=1}^{K}\sum_{j=1}^{L_k(T)}\left[R\left(f^*_{k_0}\right) - R\left(\widehat{f}_{k,j}\right)\right]\right) \\ &= \sum_{k=1}^{K} E\sum_{j=1}^{L_k(T)}\left[R\left(f^*_{k_0}\right) - R(f^*_k) + R(f^*_k) - R\left(\widehat{f}_{k,j}\right)\right] \\ &= \sum_{k=1}^{K} E[L_k(T)]\Delta_k + \sum_{k=1}^{K}\sum_{n=1}^{N} E\left[L_{k,n}(T)\right]\Delta_{k,n}. \end{aligned}$$

According to Lemma 1, the expected number of times that a sub-optimal cluster $\mathcal{F}_k$ is selected equals $O(\ln T)$. Thus for the first part, the regret is $O(K\ln T)$. For the second part, according to Lemma 2, for those sub-optimal clusters, $E[L_{k,n}(T)]$ is $O(\ln\ln T)$ for non-zero $\Delta_{k,n}$. Thus we have

$$\sum_{k\neq k_0}^{K}\sum_{n=1}^{N} E\left[L_{k,n}(T)\right]\Delta_{k,n} = o(\ln T). \tag{8}$$

In the optimal cluster $\mathcal{F}_{k_0}$, the number of times that a sub-optimal arm is pulled equals $O(\ln T)$, thus the regret in the second part is $O(N\ln T)$. By combining the two parts, we prove the theorem.

4.2. Continuous arm space

In this subsection, we develop UCB-SI2 algorithm and prove a regret bound for the continuum-armed setting. We show that for general reward function class, our proposed algorithm has a sub-linear regret bound if the complexity of the arm space is bounded.

Theorem 2. (Regret Bound for UCB-SI2 Algorithm) Denote $R\circ\mathcal{F}_k$ as the reward function class with input Y^t for cluster k, assume $R\circ\mathcal{F}_k$ has bounded pseudo dimension[3] $Pdim(R\circ\mathcal{F}_k) \le d$, for $\forall\ k$. If $\eta_{k,t}$ is set to be $6\sqrt{\frac{2d\ln T}{L_k(t)}}$, the expected regret bound for UCB-SI2 algorithm is at most $O(T^{2/3}(d\ln T)^{1/3})$.

The theorem shows if the reward function class has finite pseudo dimension, UCB-SI2 algorithm has a sub-linear regret as compared to the best arm. The proof of the theorem is based on the following lemmas.

Lemma 3. *Before any round T, the expected number of times that UCB-SI2 algorithm pulls arms from the sub-optimal cluster k is no more than* $\frac{2048d\ln T}{\Delta_k^2} + C(d)$, *where* $C(d)$ *is bounded.*

Proof. We follow the proof of Lemma 1, for any positive integer l and sub-optimal cluster $\mathcal{F}_k$, we have

$$L_k(T) \le l + \sum_{t=1}^{\infty}\sum_{s=1}^{t-1}\sum_{s_k=l}^{t-1}\left\{\widehat{R}\left(\widehat{f}_{k_0,s}, s\right) + 16\sqrt{\frac{2d\ln t}{s}} \le \widehat{R}\left(\widehat{f}_{k,s_k}, s_k\right) + 16\sqrt{\frac{2d\ln t}{s_k}}\right\}. \tag{9}$$

Let $l = \frac{2048d\ln T}{\Delta_k^2}$. When $s > l$, the indicator function in the RHS of (9) equals 1 only if at least one of the following inequalities holds,

$$R\left(f^*_{k_0}\right) - \widehat{R}\left(f^*_{k_0}, s\right) \ge 16\sqrt{\frac{2d\ln t}{s}} \tag{10}$$

$$\widehat{R}\left(\widehat{f}_{k,s_k}, s_k\right) - R\left(\widehat{f}_{k,s_k}\right) \ge 16\sqrt{\frac{2d\ln t}{s_k}}. \tag{11}$$

Applying Chernoff bound and uniform convergence bound in Theorem 29.1 in [19], the probability of (10) and (11) occur can be bounded by

$$\begin{aligned} &P\left(R\left(f^*_{k_0}\right) - \widehat{R}\left(f^*_{k_0}, s\right) \ge 16\sqrt{\frac{2d\ln t}{s}}\right) \le t^{-1024d} \\ &P\left(\widehat{R}\left(\widehat{f}_{k,s_k}, s_k\right) - R\left(\widehat{f}_{k,s_k}\right) \ge 16\sqrt{\frac{2d\ln t}{s_k}}\right) \end{aligned} \tag{12}$$

[3] Pseudo dimension is a way to measure the capacity of a set of real-valued functions. The definition of pseudo dimension can be found in [19].

$$\leq P\left(sup_{f\in F_k}|\widehat{R}(f,s_k)-R(f)|\geq 16\sqrt{\frac{2d\ln t}{s_k}}\right) \tag{13}$$

$$\leq 2\left(\sqrt{\frac{2e^2s_k}{d\ln t}}\ln\sqrt{\frac{e^2s_k}{2d\ln t}}\right)^d t^{-4d}. \tag{14}$$

Then the expected number of selecting cluster k is at most

$$E[L_k(T)]\leq \frac{2048(d\ln T)}{\Delta_k^2}+\sum_{t=1}^{\infty}\sum_{s=1}^{t-1}\sum_{s_k=1}^{t-1}3\left(\sqrt{\frac{2e^2s_k}{d\ln t}}\ln\sqrt{\frac{e^2s_k}{2d\ln t}}\right)^d t^{-4d} \tag{15}$$

$$\leq \frac{2048(d\ln T)}{\Delta_k^2}+\sum_{t=1}^{\infty}\sum_{s_k=1}^{t-1}3\left(\sqrt{\frac{2e^2s_k}{d\ln t}}\ln\sqrt{\frac{e^2s_k}{2d\ln t}}\right)^d t^{1-4d} \tag{16}$$

$$= \frac{2048(d\ln T)}{\Delta_k^2}+\sum_{t=1}^{\infty}O\left(t^{\frac{3}{4}d+1}\right)t^{1-4d}. \tag{17}$$

Note that the second term in (17) is bounded, therefore we can prove the lemma. □

Lemma 4. *Denote $G(f,\gamma)$ as a γ -radius ball around f in metric space $(\mathcal{F},d_1)$, where $d_1(f,f')=|R(f)-R(f')|$. For any cluster k, let $W(f_k^*,\gamma)=\mathcal{F}_k\backslash G(f_k^*,\gamma)$. For any round T, the expected number of times that UCB-SI2 algorithm pulls arms in $W(f_k^*,\gamma)$ is $O\left(E\left[\frac{d\ln L_k(T)}{\gamma^2}\right]\right)$.*

In order to make the gap between the best arm and the sub-optimal arms, we make a γ-radius ball around f_k^*. Pulling an arm in the ball suffers little regret while pulling one out of the ball suffers large. Similar to Lemma 2, we can prove the number of times that pulling arms in $W(f_k^*,\gamma)$ is a logarithm order of the number of times picking arms from $\mathcal{F}_k$.

Proof. Denote $L_{k,\gamma}(T)$ as the number of times that the algorithm pulls arms from $W(f_k^*,\gamma)$, and denote $\widehat{f}_{W(f_k^*,\gamma),s}$ as the arm in $W(f_k^*,\gamma)$ with the best empirical performance on s observations. Similar to Lemma 1, it is easy to show that for any positive integer l,

$$L_{k,\gamma}(T)\leq\sum_{t=1}^{L_k(T)}\left\{\widehat{R}\left(\widehat{f}_{W(f_k^*,\gamma),s},s\right)\geq\widehat{R}(f_k^*,s)\right\} \tag{18}$$

$$\leq l+\sum_{t=1}^{\infty}\sum_{s=l}^{t-1}\left\{\widehat{R}\left(\widehat{f}_{W(f_k^*,\gamma),s},s\right)\geq\widehat{R}(f_k^*,s)\right\}. \tag{19}$$

By letting $l=\frac{2048d\ln L_k(T)}{\gamma^2}$ and applying (10) and (11), we can prove the theorem. □

Now we will prove Theorem 2 based on the above lemmas.

Proof of Theorem 2. Following the proof of Theorem 1, denote $L_{k,\gamma}(T)$ as the number of times that UCB-SI2 algorithm pulls arms in $W(f_k^*,\gamma)$ before round T. Then the total regret can be decomposed by,

$$\begin{aligned}E\left(\sum_{t=1}^{T}\left[R\left(f_{k_0}^*\right)-R\left(f_{Alg}^t\right)\right]\right)&=\sum_{k=1}^{K}E[L_k(T)]\Delta_k+\sum_{k=1}^{K}E\sum_{j=1}^{L_k(T)}\left[R(f_k^*)-R\left(\widehat{f}_{k,j}\right)\right]\\&\leq\sum_{k=1}^{K}E[L_k(T)]\Delta_k+\sum_{k=1}^{K}E\left[\gamma\left[L_k(T)-L_{k,\gamma}(T)\right]+L_{k,\gamma}(T)\right]\\&\leq\sum_{k=1}^{K}E[L_k(T)]\Delta_k+\gamma T+\sum_{k=1}^{K}E\left[L_{k,\gamma}(T)\right].\end{aligned}$$

According to Lemma 3 and Lemma 4, for any sub-optimal cluster $\mathcal{F}_k$, $E[L_k(T)]$ is $O(d\ln T)$, then for the first part, the regret equals $O(Kd\ln T)$. For the second part when $k=k_0$, $E\left[L_{k_0,\gamma}(T)\right]=O\left(\frac{d\ln T}{mma^2}\right)$; For $k\neq k_0$, $E\left[L_{k_0,\gamma}(T)\right]=o(\ln T)$. By combining these parts we can obtain the regret of UCB-SI2 algorithm is

$$O(Kd\ln T)+\gamma T+O\left(\frac{d\ln T}{\gamma^2}\right)+o(\ln T). \tag{20}$$

By setting $\gamma=T^{-1/3}(d\ln T)^{1/3}$, we prove the theorem.

Theorem 2 also has its value for the continuum-armed bandit problem. It is interesting to see that the theorem gives a sub-linear regret bound for the general reward function class under conditions. In this sense, it goes beyond the results obtained by the works on continuum-armed bandit: even if the reward function is non-convex, in some other reasonable settings there may exist algorithms that can achieve a sub-linear regret bound. Furthermore, the regret bound we obtained is $O(T^{2/3}(\ln T)^{1/3})$, which is of the same order with the classical 1-d continuum-armed bandit algorithms discussed in [14]. This is very encouraging: the regret bound does not become looser when we relax the constraint on the reward function.

5. Experiment results

In this section, we introduce the experimental setting and test the effectiveness of our proposed online learning algorithms in advertising scenario. For simplicity, we assume the feature space of the ads is $\mathbb{R}^2$. In the experiment, we use linear quality score function class, i.e., $h_w(x)=<w,x>$, $w=(w_1,w_2)\in\Omega$, $\Omega=\{w:\|w\|=1, w\geqslant 0\}$, in which the parameter w can be considered as the arm or the strategy of the ad platform.

5.1. Experimental setting

We use synthetic data to verify the effectiveness of our proposed algorithms. The synthetic data is generated as follows, we sample the total number of ads M from a discrete uniform distribution in the interval [20,50]. For each advertisement i, we sample the click probability CTR_t from an uniform distribution $U(0,0.1)$, and sample the bidding price b_i from a uniform distribution U(0,100). The feature of each ad is uniformly sampled from $(0,1)\times(0,1)$.

We made experiments to test different online learning algorithms according to the above setting. For any online learning algorithm, when an arm parameter) w^t is pulled at round t, we simulate the top-L ad list according to the ranking function $s_w(x_i,b_i)=f_w(x_i)\times b_i$, as a industrial practice, we set L to be 4. If ad i is shown to the user, we simulate user's click behavior (click or not) on the ad by a Bernoulli distribution $B(1, CTR_i)$, and then the revenue of this impression can be computed and the reward of pulling arm w^t at round t is obtained.

To test the performance of USB-SI1 algorithm for multi-armed bandit setting, we first collect all potential top-L permutations that can be

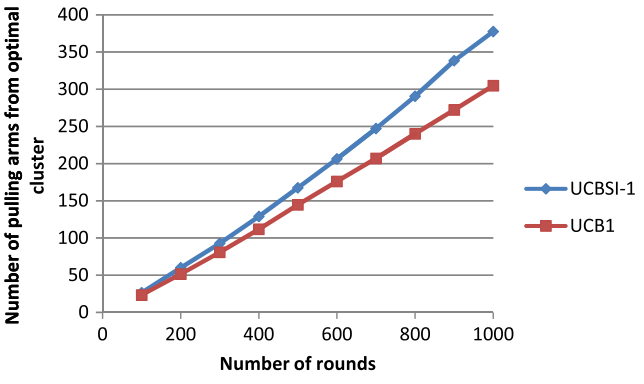

Fig. 3. The number of pulling arms from the optimal cluster by different algorithms.

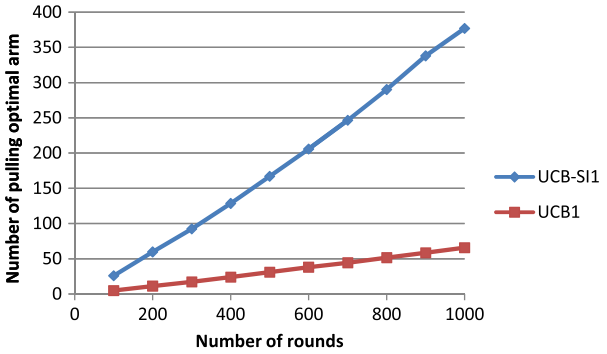

Fig. 4. The number of pulling the optimal arm by different algorithms.

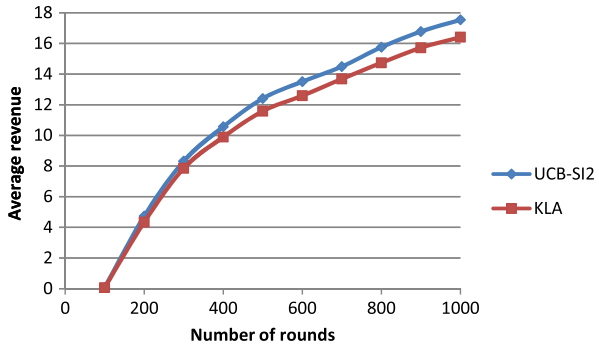

Fig. 6. Performance of UCB-SI2 and KLA.

generated from the choices of w by searching the parameter space Ω, and then randomly sample 5 different top-L permutations. For each sampled permutation we further randomly select 10 different parameter profiles that can output this top-L permutation. Thus overall, we have 50 different profiles as arms, and the arms can be divided into 5 clusters with equal sizes, then we implement USB-SI1 algorithm and use the classical UCB1 algorithm as baseline. To test the performance of USB-SI2 algorithm for continuum-armed bandit setting, we use KLA [15] algorithm as our baseline. We use simple gradient ascent in the step 5 of USB-SI1 algorithm and it is easy to check that the pseudo dimension of $R \circ \mathcal{F}_k$ used in step 6 is bounded by 4 for all k, then the proposed USB-SI2 algorithm can be implemented.

We run each of the four online learning algorithms for 1000 rounds, and run the experiment for 10 times. Average performances are reported in the next subsection.

5.2. Results

In this subsection, we report our observations and experimental results for different learning algorithms.

First we sample a set of parameters (the number of ads, *CTR* and bid of each ad) and visualize the dependency between the expected revenue and the first parameter w_1 as shown in Fig. 2. It can be easily seen from the figure that the objective function is non-continuous, non-differential and non-convex, thus it is hard for us to obtain the global maximal by general optimization methods. However, from the figure we can also see that in this case, the curve can be divided into eight pieces, and each piece actually corresponds to a top-L list. In each piece, the curve is continuous, differential and convex, which makes the optimization within a piece possible.

The experimental results of USB-SI1 and UCB1 algorithm for multi-armed bandit setting are shown in Figs 3–5. We evaluate the performance of each online learning algorithm by three factors, the number of times that the algorithm pulls arms from the optimal cluster, the number of times that the algorithm pulls the optimal arm, and the average revenue generated at different rounds. First we see from Figs. 3 and 4 that the number of times USB-SI1 algorithm pulls arms from the optimal cluster is about 15% larger than that of UCB1 algorithm, and the number of times that USB-SI1 algorithm pulls the optimal arm is about seven-times larger than that of UCB1 algorithm. This shows that first, our proposed learning strategy is better at pulling arms from the optimal cluster than UCB1 algorithm; Second, when selecting an arm from a cluster, our UCB-SI1 algorithm picks the best arm with high probability, this makes the curve of UCB-SI1 algorithm in Fig. 4 and 5 almost the same. However, for UCB1 algorithm, it still follows the upper-confidence-bound strategy within any cluster, which makes the number of pulling an optimal arm significantly smaller than UCB-SI1 algorithm. As a result, in Fig. 5, the average performance of UCB-SI1 algorithm is about 5% better than UCB1 algorithm.

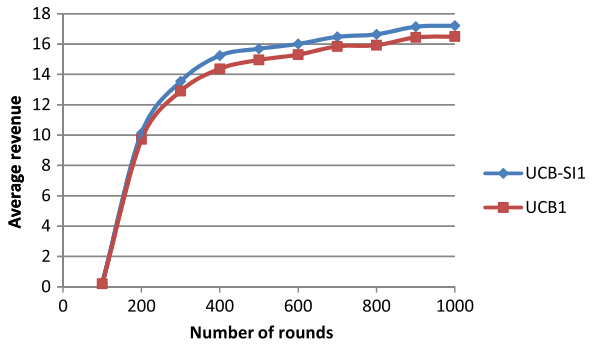

Fig. 5. The average revenue obtain by different algorithm.

The experimental results of USB-SI2 and KLA algorithm for continuum-armed bandit setting are shown in Fig. 6. It can be seen from the figure, first our UCB-SI2 algorithm outperforms the baseline KLA algorithm about 9%, this well indicates the gain of using shared information; Second, the convergence rate of both algorithm is much slower than UCB-SI1 and UCB1 algorithm, this is also consistent with our theoretical analysis.

To sum up, the experimental results are consistent with the theoretical analysis and clearly show the effectiveness of the proposed algorithms.

6. Conclusion and future work

In this paper, we have studied the online learning of auction mechanism for sponsored search. We show that this task corresponds to a new type of armed bandit problem, in which dependent arms share the explored information. We call this new problem armed bandit with shared information (AB-SI). To tackle this problem, we proposed a new algorithm called UCB-SI and prove sub-linear regret bounds in different settings.

For the future work, we plan to investigate on the following directions. First, in the paper we just analyzed the upper bound of the regret. We will study whether it can match the lower bound. Second, we assume the supervised learning algorithm is efficient to produce the empirical optimal, but the optimization error needs to be taken into consideration. Third, we will study whether our proposed new multi-armed bandit problem can be applied in other applications beyond sponsored search.

Acknowledgments

This work was supported by NSFC(61222307, 61075003) and a grant from MOE-Microsoft Laboratory of Statistics and Information Technology of Peking University. Part of this work was done when the first author visited Microsoft Research Asia.

References

[1] B. Edelman, M. Ostrovsky, M. Schwarz, Internet advertising and the generalized second-price auction: selling billions of dollars worth of keywords, The American Economic Review 97 (1) (2007) 242–259.

[2] H. Varian, Position auctions, International Journal of Industrial Organization 25 (6) (2007) 1163–1178.

[3] Y. Zhu, G. Wang, J. Yang, D. Wang, J. Yan, J. Hu, Z. Chen, Optimizing Search Engine Revenue in Sponsored Search, SIGIR 2009, ACM, 2009, 2009, pp. 588–595.

[4] F. Radlinski, A. Broder, P. Ciccolo, E. Gabrilovich, V. Josifovski, L. Riedel, Optimizing Relevance and Revenue in Ad Search: A Query Substitution Approach, SIGIR 2008, ACM, 2008, 2008, pp. 403–410.

[5] S. Lahaie, D. Pennock, Revenue Analysis of a Family of Ranking Rules for Keyword Auctions, EC 2007, ACM, 2007, 2007, pp. 50–56.

[6] D. Garg, Y. Narahari, S. Reddy, Design of an Optimal Auction for Sponsored Search Auction, E-Commerce Technology and the 4th IEEE International Conference on Enterprise Computing, E-Commerce, and E-Services, 2007. CEC/EEE 2007. The 9th IEEE International Conference on, IEEE, 2007, pp. 439–442.

[7] P. Auer, N. Cesa-Bianchi, P. Fischer, Finite-time analysis of the multiarmed bandit problem, Machine Learning 47 (2) (2002) 235–256.

[8] D. Garg, Y. Narahari, An optimal mechanism for sponsored search auctions on the web and comparison with other mechanisms, Automation Science and Engineering, IEEE Transactions on, 6(4), 2009, pp. 641–657.

[9] J. Gittins, R. Weber, K. Glazebrook, Multi-armed bandit allocation indices, vol. 25, Wiley Online Library, 1989.

[10] P. Auer, N. Cesa-Bianchi, Y. Freund, R. Schapire, Gambling in a Rigged Casino: The Adversarial Multi-Armed Bandit Problem, Foundations of Computer Science, 1995. Proceedings., 36th Annual Symposium on, IEEE, 1995, pp. 322–331.

[11] R. Gonen, E. Pavlov, An Incentive-Compatible Multi-Armed Bandit Mechanism, Proceedings of the twenty-sixth annual ACM symposium on Principles of distributed computing, ACM, 2007, pp. 362–363.

[12] S. Pandey, D. Chakrabarti, D. Agarwal, Multi-Armed Bandit Problems With Dependent Arms, Proceedings of the 24th international conference on Machine learning, ACM, 2007, pp. 721–728.

[13] R. Agrawal, The continuum-armed bandit problem, SIAM Journal on Control and Optimization 33 (1995) 1926.

[14] R. Kleinberg, Nearly tight bounds for the continuum-armed bandit problem, Advances in Neural Information Processing Systems 17 (2004) 697–704.

[15] R. Kleinberg, T. Leighton, The Value of Knowing a Demand Curve: Bounds on Regret for Online Posted-Price Auctions, Foundations of Computer Science, 2003. Proceedings. 44th Annual IEEE Symposium on, IEEE, 2003, pp. 594–605.

[16] S. Pandey, C. Olston, Handling advertisements of unknown quality in search advertising, Advances in Neural Information Processing Systems 19 (2007) 1065.

[17] M. Babaioff, Y. Sharma, A. Slivkins, Characterizing Truthful Multi-Armed Bandit Mechanisms, Proceedings of the 10th ACM conference on Electronic commerce, ACM, 2009, pp. 79–88.

[18] K. Asdemir, A dynamic model of bidding patterns in sponsored search auctions, Information Technology and Management 12 (1) (2011) 1–16.

[19] L. Devroye, L. Györfi, G. Lugosi, A probabilistic theory of pattern recognition, vol. 31, Springer Verlag, 1996.

Di He is a master student of School of EECS, Peking University. His research interests are online advertisement and machine learning.

Wei Chen is a researcher of Microsoft Research Asia. Her research interests are online advertisement and machine learning.

Liwei Wang is a professor of School of EECS, Peking University. His research interests are machine learning, and game theory.

Tie-Yan Liu is a lead researcher of Microsoft Research Asia. His research interests are online advertisement and machine learning.