A Practical Semi-Parametric Contextual Bandit

Yi Peng[1], **Miao Xie**[1*], **Jiahao Liu**[1], **Xuying Meng**[2], **Nan Li**[1], **Cheng Yang**[1], **Tao Yao**[1] and **Rong Jin**[1]

[1]Alibaba Group, Hang Zhou, China
[2]Institute of Computing Technology, Chinese Academy of Sciences
{mulong.py, xiemiao.xm, glacier.ljh, nanli.ln, charis.yangc, tao.yao, jinrong.jr}@alibaba-inc.com

Abstract

Classic multi-armed bandit algorithms are inefficient for a large number of arms. On the other hand, contextual bandit algorithms are more efficient, but they suffer from a large regret due to the bias of reward estimation with finite dimensional features. Although recent studies proposed semi-parametric bandits to overcome these defects, they assume arms' features are constant over time. However, this assumption rarely holds in practice, since real-world problems often involve underlying processes that are dynamically evolving over time especially for the special promotions like Singles' Day sales. In this paper, we formulate a novel Semi-Parametric Contextual Bandit Problem to relax this assumption. For this problem, a novel Two-Steps Upper-Confidence Bound framework, called Semi-Parametric UCB (SPUCB), is presented. It can be flexibly applied to linear parametric function problem with a satisfied gap-free bound on the n-step regret. Moreover, to make our method more practical in online system, an optimization is proposed for dealing with high dimensional features of a linear function. Extensive experiments on synthetic data as well as a real dataset from one of the largest e-commercial platforms demonstrate the superior performance of our algorithm.

1 Introduction

The multi-armed bandit problem (MAB) models an agent that simultaneously attempts to acquire new knowledge (called "exploration") and optimizes their decisions based on existing knowledge (called "exploitation"). The agent attempts to balance these competing tasks in order to maximize their total rewards over the period of time considered [Agrawal *et al.*, 1988]. Such problem is ubiquitous in many practical applications, including online advertisement [Schwartz *et al.*, 2017], personalized recommendation [Li *et al.*, 2016], medical treatment [Press, 2009] and financial portfolio design[Hoffman *et al.*, 2011]. There are often many newly emerging items with unknown rewards to be explored in these applications. Besides, MAB algorithms are very usable in cases where continuous and permanent optimization is needed for discovering the real "interests" of the customers.

MAB problem is first introduced as the sequential design of experiments, which is now formulated as a system of several candidate arms whose rewards are independently sampled from some fixed underlying distributions [Robbins, 1952]. Recently there are three typical groups of MABs: classic MABs, contextual bandits and semi-parametric bandits. Classic MAB algorithms (also called non-parametric methods), such as Upper Confidence Bound (UCB) [Auer *et al.*, 2002] and Thompson sampling [Thompson, 1933], estimate the expected reward of an arm by empirical mean of historical data. The reward estimation can be unbiased with sufficient online data leading to an inefficient convergence process for a large number of candidate arms. So it can be effective only where you can afford to wait for a long duration before knowing with certainty which decision is the best. As for contextual bandit algorithms (called parametric methods) [Chu *et al.*, 2011; Agrawal and Goyal, 2013], the learner's aim is to collect enough information about how the contextual features and rewards relate to each other modelled by a parametric function. Although they are efficient for a large arm set because the parameters can be shared among all independent candidate arms, it is generally difficult to specify a correct model and to perfectly characterize the context with numeric features, leading to large regrets in many applications [Ghosh *et al.*, 2017; Li *et al.*, 2016]. To make up shortages of the above two kinds of MABs, semi-parametric bandits, which contain both parametric part and non-parametric part, have been proposed [Ou *et al.*, 2019]. Specifically, the parametric part, which models expected reward as a parametric function of arm features, can efficiently eliminate poor arms from the candidate set whereas the non-parametric part, which adopts non-parametric model and revises the parametric estimation to avoid estimation bias. However, current semi-parametric bandits assume arms' features are constant over time. In fact, the contextual features (especially for features of users' behaviours) of candidate arms in real-world problems are usually evolved dramatically [Xie *et al.*, 2015], especially in some special promotions like Alibaba's Singles' Day sales. In these kinds of scenarios, it is required that MAB algorithms be able to adapt recommendation strategies according

*Contact Author

to the evolution of arms' contextual features. So dynamic features lead to a large regret and divergence of previous semi-parametric bandits.

In this paper, we will relax the assumption of constant features in semi-parametric bandits. It is the first time to propose a novel Semi-Parametric Contextual Bandits Problem, as far as we known, for cases where contextual features are evolved dramatically. To solve it, a novel Two-Steps Upper-Confidence Bound framework, called Semi-Parametric UCB (SPUCB), is presented. It can be flexibly applied to Top-1 and Top-K selection problems. We give the implementation of utilizing a linear function as the parametric part and prove satisfied gap-free bounds on the n-step regret. Extensive experiments show that SPUCB leads to significantly higher qualified arms because it can take dynamic contextual features in semi-parametric environment.

2 Related Work

Recently there are three typical groups of MABs: classic MABs [Auer *et al.*, 2002; Gopalan *et al.*, 2014; Thompson, 1933], contextual bandits [Agrawal and Goyal, 2013; Chu *et al.*, 2011; Qin *et al.*, 2014; Li *et al.*, 2016] and semi-parametric bandits [Ou *et al.*, 2019; Ghosh *et al.*, 2017; Krishnamurthy *et al.*, 2018; Greenewald *et al.*, 2017]. Classic MAB algorithms can be called non-parametric methods and contextual bandits are usually seen as parametric methods. In addition, to solve top-k combinatorial optimization problem, combinatorial variants [Gai *et al.*, 2012; Chen *et al.*, 2013; Kveton *et al.*, 2015a; Li *et al.*, 2016] are proposed.

Firstly, classic MAB algorithms like UCB and Thompson sampling are inefficient when the arm set is large because they have to explore each arm independently. Secondly, to leverage contexts, Li *et al.* [2010] formulated personalized news recommendation as a contextual bandit problem. They proposed LinUCB to maximize total clicks based on contextual features about the users and articles. Qin *et al.* [2014] proposed C2-UCB, a contextual combinatorial bandit algorithm, which considers semi-bandit feedback and non-linear reward. Li *et al.* [2016] combined contextual cascading bandit with position discounts. Their C^3-UCB algorithm is bounded by $O(d\sqrt{KT}\log T)$. Although they are efficient for a large arm set because parameters can be shared among all independent candidate arms, it is generally difficult to specify a correct model and to perfectly characterize the context with numeric features. It makes the assumption impractical, leading to a large regret. Their methods divergent for Semi-Parametric Contextual Bandit Problem due to the latent arm bias. To make up the shortages of above two kinds of MABs, semi-parametric bandits, which contain both parametric part and non-parametric part, have been proposed [Ghosh *et al.*, 2017; Ou *et al.*, 2019; Krishnamurthy *et al.*, 2018]. Krishnamurthy *et al.* [2018] assumed the reward is a summation of a linear function of context features and an arm-independent bias. Ghosh *et al.* [2017] assumed each arm has an arm-specific bias which is more flexible in practice. Ou *et al.* [2019] proposed a novel framework, called Semi-Parametric Sampling, which can work in the semi-parametric environment. However, these recent methods for semi-parametric bandit problems assume arms' features are constant over the time, so they will suffer from a large regret in our defined problem. Thirdly, combinatorial bandits are presented for top-k combinatorial optimization problem. Generic stochastic combinatorial bandits are pioneered by Gai *et al.* [2012], who proposed a UCB-like algorithm, named LLR. Kveton *et al.* [2015a] introduced the *cascading model* to combinatorial semi-bandits and achieved an upper bound of $O(\frac{L}{\Delta_{\min}}\log T)$. Kveton *et al.* [2015b] proposed a computationally efficient and sample efficient framework, called CombCascade, allowing each feasible action to be a subset of ground items under combinatorial constraints. Its regret is bounded by $O(\frac{KL}{\Delta_{\min}}\log T)$. Li *et al.* [2016] proposed contextual combinatorial cascading with position discounts. However, none of them can be used for our problem directly.

3 Semi-Parametric Contextual Bandit

Suppose there is a tuple $B = (\mathcal{E}, K, \mathcal{S})$, where $\mathcal{E} = [L]$ is a ground set of L items (also referred to as candidate arms), integer $K \geq 1$ and $\mathcal{S}$ is the set of all K-tuples of distinct items coming from $\mathcal{E}$. The contextual features of candidate arms usually evolved dramatically which reflect the user's dynamic real interests, a novel ***Semi-Parametric Contextual Bandit Problem*** should be defined as follows:

Let T be the length of time horizon. At time t, the learning agent observes a feature vector $x_{t,a} \in \mathbb{R}^{d\times 1}$ with $\|x_{t,a}\|_2 \leq 1$ for every candidate arm $a \in \mathcal{E}$, and recommends to the user a list of K items $A_t = (a_1^t, \cdots, a_K^t) \in \mathcal{S}$. Users will give feedback (reward) to the recommended list. Let $\mathcal{H}_t$ denotes the history within T. It contains rewards and contextual information before the agent chooses an action at time t. We assume that each base arm a has an independent weight $\mathbf{w}_t(a)$, representing the "quality" of a at time t. Given history $\mathcal{H}_t$, expected reward can be formulated with a semi-parametric form:

$$w_{t,a}^* = \mathbb{E}[\mathbf{w}_t(a)|\mathcal{H}_t] = g(\theta^*, x_{t,a}) + b_a^*, \tag{1}$$

where $g(\theta^*, x_{t,a})$ is a parametric reward function of arm features $x_{t,a}$ at time t, $b_a^* \in \mathbb{R}$ is the bias of parametric function g and real expected reward $w_{t,a}^* in [0,1]$, θ^* is the optimal function parameter defined as

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \sum_{t=1}^{T}\sum_{a=1}^{L} |(\mathbf{w}_t(a) - b_a^*) - g(\theta, x_{t,a})|^2. \tag{2}$$

For expected reward $w_{t,a}^*$, we call b_a^* the non-parametric part and θ^* the parametric part, because b_a^* are independent among arms with no parametric assumption. θ^* is an unknown d-dimensional vector with the assumption that $\|\theta^*\|_2 \leq 1$ for all t, a.

We assume that each weight $\mathbf{w}_t(a)$ follows R_α-sub-Gaussian tail $\eta_{t,a} = \mathbf{w}_t(a) - w_{t,a}^*$, which means

$$\exists R_\alpha > 0 \quad \forall \mu \in \mathbb{R} \quad \mathbb{E}[e^{\mu\eta_{t,a}}|\mathcal{H}_t] \leq e^{R_\alpha^2\mu^2/2}.$$

The sub-Gaussian condition implies that $\eta_{t,a}$ is a zero-mean Gaussian noise with variance at most R_α^2 lying in an interval of length at most $2R_\alpha$.

The objective of Semi-Parametric Contextual Bandit Problem is to find a policy that can maximize the cumulative expected reward. Since direct analysis of cumulative reward is not tractable, we analyze the cumulative regret.

In the case of $K = 1$, the user only chooses the best item, the cumulative regret is defined as:

$$\mathrm{R}(t) = \sum_{t=1}^{T} w_{t,a}^* - \sum_{t=1}^{T} \mathbf{w}_t(a)$$

When $K > 1$, the problem will be a combinatorial optimization problem which can be reduced to a set cover problem in each time slot, thus it is a NP-hard problem. To propose an approximate solution, we use the cascading model to reduce it to a ranking problem with the following oracle.

In the cascading model, the user examines items in a presented order and stops at the $\mathbf{O}_t$-th item when he finds the first attractive one. Then the agent will give a positive reward for $\mathbf{O}_t$-th arm, while put a negative reward for the others in A_t. It is required that the decision strategy, denoted as $f(A, w)$, for recommended arms A based on the given weights w satisfy monotonicity and B-Lipschitz continuity assumptions. We only assume that the agent has access to an α-approximation oracle $\mathcal{O}_{\mathcal{S}}(w)$ that outputs recommended action A, which means given input w, for some $\alpha \leq 1$, the oracle returns an action $A = \mathcal{O}_{\mathcal{S}}(w) \in \mathcal{S}$ satisfying $f(A, w) \geq \alpha f(A^*, w)$ where $A^* = \mathrm{argmax}_{A \in \mathcal{S}} f(A, w)$.

The α-regret of action A on time t is

$$\mathrm{R}^{\alpha}(t, A) = \alpha f_t^* - f(A, w_t),$$

where $f_t^* = f(A_t^*, w_t)$ and $w_t = (g(\theta^*, x_{t,a}) + b_*)_{a \in \mathcal{E}}$.

Thus the cumulative regret of an algorithm in the case of $K > 1$ is to minimize the α-regret

$$\mathrm{R}^{\alpha}(T) = \mathbb{E}[\sum_{t=1}^{T} \mathrm{R}^{\alpha}(t, A_t)].$$

4 Two-Steps UCB Framework

We introduce a novel Two-Steps Upper-Confidence Bound framework, called Semi-Parametric UCB (SPUCB) to solve Semi-Parametric Contextual Bandit Problem, our method is unbiased and efficient.

From Eq.(1), θ^* and b_a^* should be inferred according to the rewards that the learning agent can only observe within T steps. SPUCB framework owns the efficiency of parametric bandit and the unbiased property of non-parametric bandit. The parametric part helps to provide a prior of expected reward with the latest contextual information to rapidly reduce the candidate arm set. Thus the non-parametric part can work in a relatively small candidate set to efficiently estimate the expected reward and correctly discover the optimal arm.

SPUCB is a UCB-like algorithm, which operates in two stages. Firstly, the learning agent computes the upper confidence bounds (UCBs) $\mathbf{U}_t$ on the expected weights of all base arms in $\mathcal{E}$, and uses them to select an action $A_t = (a_1^t, \cdots, a_K^t)$. Secondly, after the agent obtained the rewards $\mathbf{w}_t$, it use the training samples to get a new estimate $\hat{\theta}_t$ of θ^* with a new confidence radius, and estimate b_a^* based on $\hat{\theta}_t$.

For convenience, we define $s(t, a)$ to be the number of times that candidate arm a is observed in $t - 1$ steps.

Lemma 4.1. *Let $\alpha_t(a) = R_\alpha \sqrt{-2\log(\delta/2)/s}$. With probability at least $1 - \delta$, for each arm a at time $t > 0$, we have:*

$$|\hat{\mathbf{w}}_s(a) - g_s(\theta^*, x_a) - b_a^*| \leq \alpha_t(a), \tag{3}$$

where $\hat{\mathbf{w}}_s(a)$ is the average of s observed weights of candidate arm a, $g_s(\theta, x_a) = \frac{1}{s}\sum_{\tau=1}^{s} g(\theta, x_{\tau,a})$ is the average of g calculated from observed feature vectors of a and θ.

Proof. Let $w_{s,a}^* = \frac{1}{s}\sum_{r=1}^{s} w_{r,a}^*$, we have $b_a^* = \frac{1}{s}\sum_{r=1}^{s}\left(w_{r,a}^* - g(\theta^*, x_{r,a})\right) = w_{s,a}^* - g_s(\theta^*, x_a)$ by Eq.(1). Because η is a zero-mean Gaussian noise lying in $[-R_\alpha, R_\alpha]$, by Hoeffding inequality, we have

$$\begin{aligned}
&\mathbb{P}\left(|\hat{\mathbf{w}}_s(a) - g_s(\theta^*, x_a) - b_a^*| \leq \alpha_{t-1}(a)\right) \\
=&\mathbb{P}\left(\left|\frac{1}{s}\sum_{r=1}^{s}\eta_{r,a}\right| \leq \alpha_{t-1}(a)\right) \\
\geq&1 - 2\exp\left(-\frac{2s^2\alpha_{t-1}^2(a)}{\sum_{r=1}^{s}(R-(-R))^2}\right) = 1 - \delta.
\end{aligned} \tag{4}$$

□

Lemma 4.2. *Given*

$$\left|g(\hat{\theta}_t, x_{t,a}) - g(\theta^*, x_{t,a}) - g_s(\hat{\theta}, x_a) + g_s(\theta^*, x_a)\right| \leq \gamma_t(a).$$

Then, with probability at least $1 - \delta$, we have:

$$\left|\hat{\mathbf{w}}_s(a) - g_s(\hat{\theta}, x_a) + g(\hat{\theta}_t, x_{t,a}) - w_{t,a}^*\right| \leq \ \alpha_t(a) + \gamma_t(a). \tag{5}$$

Proof. Based on Lemma 4.1 and Eq.(1), we have

$$\begin{aligned}
&\left|\hat{\mathbf{w}}_s(a) - g_s(\hat{\theta}, x_a) + g(\hat{\theta}_t, x_{t,a}) - w_{t,a}^*\right| \\
=&\left|\hat{\mathbf{w}}_s(a) - g_s(\hat{\theta}, x_a) + g(\hat{\theta}_t, x_{t,a}) - g(\theta^*, x_{t,a}) - b_a^*\right| \\
=&\Big|\hat{\mathbf{w}}_s(a) - g_s(\theta^*, x_a) - b_a^* + g(\hat{\theta}_t, x_{t,a}) \\
&-g(\theta^*, x_{t,a}) - g_s(\hat{\theta}, x_a) + g_s(\theta^*, x_a)\Big| \leq \alpha_t(a) + \gamma_t(a).
\end{aligned} \tag{6}$$

□

Based on Lemma 4.1 and Lemma 4.2, the upper confidence bound of expected weight for each candidate arm a at time t can be defined as

$$\mathbf{U}_t(a) = \hat{\mathbf{w}}_s(a) - g_s(\hat{\theta}, x_a) + g(\hat{\theta}_t, x_{t,a}) + \alpha_t(a) + \gamma_t(a). \tag{7}$$

Thus, according to Eq.(7), SPUCB can be presented in Algorithm 1. As soon as we give the specific approach for inferring and updating $\hat{\theta}_t$ and $\gamma_t(a)$, SPUCB will be a feasible solution for Semi-Parametric Contextual Bandit problem. In Section 5, we will implement SPUCB with linear parametric function and give a satisfied gap-free bound on the T-step regret.

Algorithm 1 SPUCB

// Initialization
Observe $x_0, \mathbf{w}_0 \sim P$
$s_0 \leftarrow 1, \hat{\theta}_0 \leftarrow 0, \alpha_0 \leftarrow R_\alpha\sqrt{-2\log(\delta/2)}, \gamma_0 \leftarrow \alpha_0$
$\hat{\mathbf{w}}_1 \leftarrow \mathbf{w}_0,\ x_1 \leftarrow x_0$
for all $t = 1, \cdots, T$ **do**
  // Compute UCB
  for all $a = 1, \cdots, L$ **do**
    $\mathbf{U}_t(a) \leftarrow \hat{\mathbf{w}}_{s_{t-1}(a)} - g_{s_{t-1}(a)}(\hat{\theta}, x_a) + g(\hat{\theta}_{t-1}, x_{t,a}) + \alpha_t(a) + \gamma_t(a)$
  // Recommend and observe
  $A_t = (a_1^t, \cdots, a_K^t) \leftarrow \mathcal{O}_S(\mathbf{U}_t)$
  Play A_t and observe $\mathbf{O}_t, \mathbf{w}_t(a)$
  // Update statistics
  Compute $\hat{\theta}_t$ using x and $\mathbf{w}$
  for all $a = 1, \cdots, L$ **do**
    $s_t(a) \leftarrow s_{t-1}(a)$
  for all $k = 1, \cdots, \mathbf{O}_t$ **do**
    $a \leftarrow a_k^t$
    $s_t(a) \leftarrow s_t(a) + 1$
    $\hat{\mathbf{w}}_{s_t(a)} \leftarrow \frac{s_{t-1}(a)\hat{\mathbf{w}}_{s_{t-1}(a)} + \mathbf{w}_{s_t(a)}(a)}{s_t(a)}$
    Compute $g_{s_t(a)}$ using $\hat{\theta}$ and x_a
  for all $a = 1, \cdots, L$ **do**
    $\alpha_t(a) \leftarrow R_\alpha\sqrt{-2\log(\delta/2)/s_t(a)}$
    Compute $\gamma_t(a)$

5 Linear Parametric Implementation

We introduce SPUCB to solve Top-1 ($K = 1$) and Top-K ($K \geq 2$) semi-parametric contextual bandit problems with linear parametric functions and achieved gap-free bounds given on the T-step regret. When the parametric part of Eq.(1) is a linear function, we have the following expected reward at time t,

$$w_{t,\mathbf{a}_k^t} = g(\theta^*, x_{t,a}) + b_* = \theta_*^T x_{t,\mathbf{a}_k^t} + b_*.$$

We estimate θ_* using a ridge regression on samples Δx_t and obtained rewards $\Delta\mathbf{w}$, then we can get a l^2-regularized least-squares estimate with a regularization parameter $\lambda > 0$:

$$\hat{\theta}_t = (\Delta\mathbf{X}_t^T\Delta\mathbf{X}_t + \lambda\mathbf{I})\Delta\mathbf{X}_t^T\Delta\mathbf{Y}_t,$$

where $\Delta\mathbf{X}_t \in \mathbb{R}^{(\sum_{\tau=1}^t \mathbf{O}_\tau)\times d}$ is the matrix whose rows are $\Delta x_{\tau,a}^T = x_{\tau,a}^T - \frac{1}{s(\tau,a)}\sum_{r=1}^{s(\tau,a)} x_{r,a}^T$ and $\Delta\mathbf{Y}_t$ is a column vector whose elements are $\Delta\mathbf{w}_\tau(a) = \mathbf{w}_\tau(a) - \frac{1}{s(\tau,a)}\sum_{r=1}^{s(\tau,a)}\mathbf{w}_r(a)$, $a = \mathbf{a}_k^\tau, k \in [\mathbf{O}_\tau], \tau \in [t]$. Let

$$\Delta\mathbf{V}_t = \lambda\mathbf{I} + \sum_{\tau=1}^{t}\sum_{k=1}^{\mathbf{O}_\tau}\Delta x_{\tau,\mathbf{a}_k^\tau}\Delta x_{\tau,\mathbf{a}_k^\tau}^T,$$

which is a symmetric positive definite matrix.

To analyze $\gamma_t(a)$, we construct an index set $\Psi_t \subseteq \{1, \cdots, t-1\}$ so that for fixed $\Delta x_{\tau,a}$ with $\tau \in \Psi_t$, $\Delta\mathbf{w}_\tau(a)$ are independent random variables [Auer, 2002]. Then we can have Lemma 5.1.

Lemma 5.1. *With probability at least $1 - \delta/T$, we have*

$$\left|g(\hat{\theta}_t, x_{t,a}) - g(\theta^*, x_{t,a}) - g_s(\hat{\theta}, x_a) + g_s(\theta^*, x_a)\right| \leq \gamma_t(a), \tag{8}$$

where $\gamma_t(a) = (R_\gamma + 1)\left\|\Delta x_{t,a}\right\|_{\Delta\mathbf{V}_{t-1}^{-1}}$ and $\|x\|_V = (x^T V x)^{\frac{1}{2}}$ is the 2-norm of x based on V.

Proof. The proof is similar to the proof of [Lemma 1, Chu *et al.*]. □

Then we can define an upper confidence bound of expected weight for each base arm a by

$$\mathbf{U}_t(a) = \min\{\hat{\mathbf{w}}_s(a) + \hat{\theta}_{t-1}^T\Delta x_{t,a} + \alpha_t(a) + \gamma_t(a), 1\}. \tag{9}$$

5.1 Regret Analysis in Top-1 Scenario

We follow the same method as [Auer *et al.*, 2002] for choosing Ψ_t carefully to achieve the independence condition.

Lemma 5.2. *Assuming $|\Psi_{T+1}| \geq 2$, we have*

$$\sum_{t\in\Psi_{T+1}}\left\|\Delta x_{t,a}\right\|_{\Delta\mathbf{V}_{t-1}^{-1}} \leq 5\sqrt{d\,|\Psi_{T+1}|\ln|\Psi_{T+1}|}. \tag{10}$$

Proof. The proof is similar to the proof of [Lemma 3, Chu *et al.*]. □

Theorem 5.1. *If* SPUCB *runs with $R_\gamma = \sqrt{\frac{1}{2}\ln\frac{2LT}{\delta}}$, then with probability at least $1 - \delta$, the regret of the algorithm is*

$$\mathrm{O}\left(\sqrt{dT\ln^3(LT\ln(T)/\delta)} + \sqrt{LT\ln(1/\delta)}\right). \tag{11}$$

Proof. Because

$$\sum_{t\in\Psi_{T+1}}\alpha_t(a) \leq \mathrm{O}\left(\sqrt{L\,|\Psi_{T+1}|\ln(1/\delta)}\right),$$

the proof is similar to the proof of [Theorem 1, Chu *et al.*]. □

5.2 Regret Analysis in Top-k Scenario

We use a cascading model to reduce the problem to a ranking problem with an offline oracle [Li *et al.*, 2016].

In the problem of cascading recommendation, when recommended with an ordered list of items $\mathbf{A}_t = (\mathbf{a}_1^t, \cdots, \mathbf{a}_K^t)$, the user examines the items in that order. The examination process will stop if the user clicks one item or has examined all items without any click. The weight of each base arm a at time t, $\mathbf{w}_t(a) \in \{0, 1\}$, indicates whether the user has clicked an item or not. Then the random variable $\mathbf{O}_t$ satisfies

$$\mathbf{O}_t = \begin{cases} k, & \text{if } \mathbf{w}_t(\mathbf{a}_j^t) = 0, \forall j < k \text{ and } \mathbf{w}_t(\mathbf{a}_k^t) = 1; \\ K, & \text{if } \mathbf{w}_t(\mathbf{a}_j^t) = 0, \forall j \leq K. \end{cases}$$

If the user clicks the k-th item, the learning agent will receive a reward 1; otherwise, there is no reward. At the end of step t, the learning agent observes $\mathbf{w}_t(\mathbf{a}_k^t), k \in [\mathbf{O}_t]$ and receives a reward $r_t = \max_{1\leq k\leq K}\mathbf{w}_t(\mathbf{a}_k^t)$. Notice that the order of $\mathbf{A}_t$ affects both the feedback and the reward.

Let us define a function $f : \mathcal{S} \times [0,1]^{|\mathcal{E}|} \to [0,1]$ on $A = (a_1, \cdots, a_K) \in \mathcal{S}, w = (w(1), \cdots, w(L))$ by

$$f(A, w) = \sum_{k=1}^{K} \prod_{i=1}^{k-1} (1 - w(a_i)) w(a_k). \tag{12}$$

It is easy to verify that let $r_t = f(A, w)$, we have $\mathbb{E}[r_t] = f(A, w)$, where f is the expected reward function and also a function of expected weights. Moreover, f satisfies the properties of monotonicity and B-Lipschitz continuity by [Lemma B, Li *et al.*] with $\gamma_k = 1, k \in [K]$. Let $A^* = \text{argmax}_{A \in \mathcal{S}} f(A, w)$ be the optimal solution, then $\Delta_A = f(A^*, w) - f(A, w)$ is a suboptimality gap of solution A, and the probability that all items in A are observed is $p_{t,A} = \prod_{k=1}^{K-1}(1 - w(a_k))$. For regret analysis, we define shorthands $p^* = \min_{1 \le t \le T} \{\min_{A \in \mathcal{S}} p_{t,A}\}$.

Lemma 5.3. *For any time t and $\mathbf{A}_t = (\mathbf{a}_1^t, \cdots, \mathbf{a}_k^t)$, if f satisfies the assumptions of monotonicity and B-Lipschitz continuity, we have*

$$\mathrm{R}^{\alpha}(t, \mathbf{A}_t) \le 2B \sum_{k=1}^{K} \left(\alpha_t(\mathbf{a}_k^t) + \gamma_t(\mathbf{a}_k^t)\right).$$

Proof. Because

$$f(\mathbf{A}_t, \mathbf{U}_t) \ge \alpha f(A^*, \mathbf{U}_t) \ge \alpha f(A_t^*, \mathbf{U}_t) \ge \alpha f(A_t^*, w_t),$$

we have

$$\begin{aligned} \mathrm{R}^{\alpha}(t, \mathbf{A}_t) &= \alpha f(A_t^*, w_t) - f(\mathbf{A}_t, w_t) \\ &\le f(\mathbf{A}_t, \mathbf{U}_t) - f(\mathbf{A}_t, w_t). \end{aligned}$$

By Lipschitz continuity of f and Lemma 4.2,

$$\begin{aligned} \mathrm{R}^{\alpha}(t, \mathbf{A}_t) &\le B \sum_{k=1}^{K} \left| \mathbf{U}_t(\mathbf{a}_k^t) - w_{t, \mathbf{a}_k^t} \right| \\ &\le 2B \sum_{k=1}^{K} \left(\alpha_t(\mathbf{a}_k^t) + \gamma_t(\mathbf{a}_k^t)\right). \end{aligned}$$

□

Notice that the upper bound of $\mathrm{R}^{\alpha}(t, \mathbf{A}_t)$ is in terms of all base arms of $\mathbf{A}_t$. However, because $\Delta \mathbf{V}_{t-1}^{-1}$ only contains information of observed base arms, it is hard to estimate an upper bound for $\sum_{t=1}^{T} \mathrm{R}^{\alpha}(t, \mathbf{A}_t)$.

Lemma 5.4. *Suppose Ineq.(3) holds for time $1 \le t \le T$.*

$$\mathbb{E}\left[\sum_{t=1}^{T} \mathrm{R}^{\alpha}(t, \mathbf{A}_t) \middle| \mathcal{H}_t\right] \le \frac{2B}{p^*} \mathbb{E}\left[\sum_{t=1}^{T} \sum_{k=1}^{\mathbf{O}_t} \left(\alpha_t(\mathbf{a}_k^t) + \gamma_t(\mathbf{a}_k^t)\right)\right]. \tag{13}$$

Proof.

$$\begin{aligned} &\mathbb{E}\left[\sum_{t=1}^{T} \mathrm{R}^{\alpha}(t, \mathbf{A}_t) \middle| \mathcal{H}_t\right] \\ =&\mathbb{E}\left[\sum_{t=1}^{T} \mathrm{R}^{\alpha}(t, \mathbf{A}_t) \mathbb{E}\left[\frac{1}{p_{t, \mathbf{A}_t}} 1\{\mathbf{O}_t = K\} \middle| \mathbf{A}_t\right] \middle| \mathcal{H}_t\right] \\ \le&\frac{1}{p^*} \mathbb{E}\left[\sum_{t=1}^{T} \mathrm{R}^{\alpha}(t, \mathbf{A}_t) 1\{\mathbf{O}_t = K\} \middle| \mathcal{H}_t\right] \\ \le&\frac{2B}{p^*} \mathbb{E}\left[\sum_{t=1}^{T} \sum_{k=1}^{\mathbf{O}_t} \left(\alpha_t(\mathbf{a}_k^t) + \gamma_t(\mathbf{a}_k^t)\right)\right]. \end{aligned}$$

□

Lemma 5.5. *Assuming $|\Psi_{T+1}| \ge 2$, we have*

$$\sum_{t \in \Psi_{T+1}} \sum_{k=1}^{\mathbf{O}_t} \alpha_t(\mathbf{a}_k^t) \le \mathrm{O}\left(\sqrt{KL\,|\Psi_{T+1}| \ln(1/\delta)}\right).$$

Proof.

$$\begin{aligned} \sum_{t \in \Psi_{T+1}} \sum_{k=1}^{\mathbf{O}_t} \alpha_{t-1}(\mathbf{a}_k^t) \le& \sqrt{2\log(2/\delta)} \sum_{t \in \Psi_{T+1}} \sum_{k=1}^{K} \frac{1}{\sqrt{s(t, \mathbf{a}_k^t)}} \\ \le& \sqrt{2\log(2/\delta)} \sum_{t \in \Psi_{T+1}} K \cdot \frac{1}{\sqrt{tK/L}} \\ \le& \mathrm{O}\left(\sqrt{KL\,|\Psi_{T+1}| \ln(1/\delta)}\right). \end{aligned}$$

□

Theorem 5.2. *If* SPUCB *runs with $R_\gamma = \sqrt{\frac{1}{2} \ln \frac{2LT}{\delta}}$, then with probability at least $1 - \delta$, the regret of the algorithm is*

$$\mathrm{O}\left(\sqrt{dKT \ln^3(LT \ln(T)/\delta)} + \sqrt{KLT \ln(1/\delta)}\right).$$

Proof. Based on Lemma 5.3 and Lemma 5.5, the proof is similar to the proof of [Theorem 1, Chu *et al.*]. □

Although the upper bound $\tilde{\mathrm{O}}(\sqrt{T})$ of SPUCB for Top-K problem is the same as C^3-UCB, our method can accurately find the optimal item set and outperform C^3-UCB because the non-parametric part of Semi-Parametric Contextual Bandit Problem will cause a large regret for C^3-UCB, which is also shown in the experiments in Section 6.

5.3 Online Optimization

To make our method more practical in real production system, an optimization technique on the linear parametric term is proposed to deal with high dimensional features.

SPUCB needs to calculate $\|\Delta x_{t,a}\|_{\Delta \mathbf{V}_{t-1}^{-1}}$ in each step for all arms leading to an $O(d^3)$ time complexity, where d is the dimension of feature vector. To overcome the bottleneck of matrix inversion, we use a novel iterative formula to reduce the complexity to $O(d^2)$ without any accuracy loss.

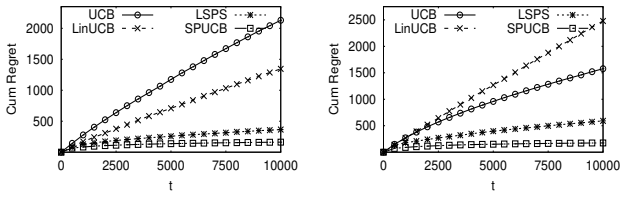
Because of $\Delta \mathbf{V}_{t+1} = \Delta \mathbf{V}_t + \Delta x_{t, \mathbf{a}_k^t} \Delta x_{t, \mathbf{a}_k^t}^T$, thus

$$\Delta \mathbf{V}_{t+1}^{-1} \leftarrow \Delta \mathbf{V}_t^{-1} - \frac{\Delta \mathbf{V}_t^{-1} \Delta x_{t, \mathbf{a}_k^t} \Delta x_{t, \mathbf{a}_k^t}^T \Delta \mathbf{V}_t^{-1}}{\Delta x_{t, \mathbf{a}_k^T}^T \Delta \mathbf{V}_t^{-1} \Delta x_{t, \mathbf{a}_k^t} + 1} \tag{14}$$

6 Experiments

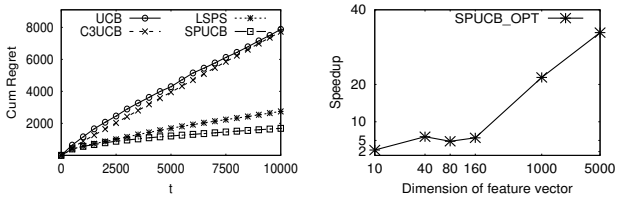
SPUCB's performance and efficiency will be evaluated by extensive experiments on both synthetic and real-world data sets, compared with the following state-of-art algorithms:

- UCB [Auer *et al.*, 2002]: a representative classic MAB, which can be seen as a pure non-parametric model.
- LINUCB [Chu *et al.*, 2011]: a representative contextual bandit, which can be seen as a pure parametric model.
- C^3-UCB [Li *et al.*, 2016]: a representative contextual bandit algorithm with a cascading model, which can be seen as a pure parametric model used in Top-K problem.



(a) $a = 0.6$ (b) $a = 0.4$

Figure 1: Cumulative regret comparison on synthetic dataset for $K = 1, L = 100, d = 10$.



(a) Cumulative regret (b) Efficiency improvement

Figure 2: Regret comparison for $K = 5, a = 0.5$ and efficiency improvement on synthetic dataset.

- LSPS [Ou *et al.*, 2019]: a representative semi-parametric bandit algorithm using Thompson Sampling.

All super-parameters of the above algorithms are tuned by a cross-validation experiment with the best performance. (Specifically, UCB: $c = 0.2$, LINUCB: $\alpha = 3.5$, LSPS: $\sigma_1 = 0.3, \sigma_2 = 0.01, \sigma_3 = 0.3$, SPUCB: $R = 0.2, \delta = 0.9, \lambda = 1.0, R_r = 0.5$)

6.1 Experiments on Synthetic Data

The synthetic dataset is randomly generated following our assumptions. Firstly, we sample linear parameters from a standard Gaussian distribution and normalize them so that their norms are equal to a predefined constant $a \in [0, 1]$. Secondly, the bias of each arm is uniformly sampled from $[0, 1 - a]$. So a can be used to control the weight of non-parameter part. Thirdly, in each time step, contextual features of each arm are sampled from a standard Gaussian distribution and are normalized to $[0, 1]$. The stochastic reward of an arm is sampled from a Gaussian distribution whose expectation is the expected reward of the arm calculated by Eq. 1 in the situation of Top-1 ($K = 1$). In Top-K ($K \geq 2$) scenario, we need to simulate the cascading model. The stochastic reward of an arm is sampled from a binomial distribution whose positive probability is its expected reward (by Eq. 1). Specifically, We run algorithms for $10,000$ time steps at most and set $L = 100, d = 10, a = 0.4$. All experiments are repeated for 50 times and reported average results.

The performance comparison on average cumulative regrets is shown in Figure 1 when $K = 1$. Firstly, from the curve, we can find that all algorithms except SPUCB failed to converge within $10,000$ time steps. The performance of SPUCB is significantly superior to other algorithms. Secondly, Although UCB's coverage speed will be faster if we enlarge the weight of non-parametric part, the cumulative regrets are still worse than that of SPUCB, shown in Figure 1(b). On the contrary, LINUCB will be better if we set a small weight for non-parametric part, shown in Figure 1(a). Thirdly, even if LSPS is better than UCB and LINUCB, its assumption that arms' features are constant over time leading to an increasing gap in cumulative regrets compared with SPUCB. Finally, SPUCB can also converge quickly and still be superior to the others in the case of $K = 5$, which is shown in Figure 2(a). Although the regret upper bound of SPUCB is the same with C^3-UCB, SPUCB outperforms C^3-UCB because of the non-parametric part of Semi-Parametric Contextual Bandit Problem. Moreover, the achieved speed-up of the proposed online optimization of Section 5.3 for SPUCB is given in Figure 2(b). The optimization will significantly improve its efficiency for nearly 34 times when the dimension of feature vector, d, is 5000. The total running time for online updating parameters is less than 20ms with a single machine with 4-core CPUs. These results illustrate that the proposed optimization technique makes our algorithm more practical.

6.2 Experiments on Real-World Data

The real-world dataset is collected from one of the largest e-commercial platform in China for the problem of products recommendation. It contains 100 items with 42-dimensional features from a special holiday. The features of arms contain several statistical features which evolve greatly over time. To make the experiments reproducible, we train a DNN model based on historical user behaviours on the items to simulate the reward function, and use it as the environment to evaluate online algorithms. In each time step, stochastic reward of an arm is sampled from a binomial distribution whose expectation is the expected reward of the arm. Figure 3 shows the comparison of cumulative regrets. As linear reward function is a mis-specified function, LINUCB suffers large regrets. Besides, LSPS is also worse than SPUCB because it cannot deal with evolving features properly.

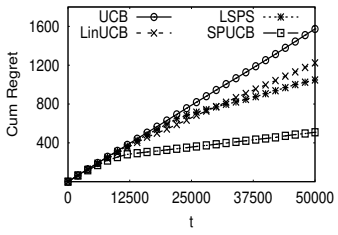


Figure 3: Regret comparison on real-world dataset.

7 Conclusions

We study a novel Semi-Parametric Contextual Bandit Problem and propose a UCB-like framework SPUCB to deal with incomplete and evolving features. It can be flexibly implemented to solve various problems with parametric functions, and has a gap-free bound. Moreover, an optimization technique is proposed to handle high dimensional features with a linear parametric function. Our method has also been deployed as a service to support online businesses in Alibaba.

References

[Agrawal and Goyal, 2013] Shipra Agrawal and Navin Goyal. Thompson sampling for contextual bandits with linear payoffs. In *International Conference on Machine Learning*, pages 127–135, 2013.

[Agrawal *et al.*, 1988] R. Agrawal, M. V. Hedge, and D. Teneketzis. Asymptotically efficient adaptive allocation rules for the multiarmed bandit problem with switching cost. *IEEE Transactions on Automatic Control*, 33(10):899–906, Oct 1988.

[Auer *et al.*, 2002] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-Time Analysis of the Multiarmed Bandit Problem. *Machine Learning*, 47(2-3):235–256, 2002.

[Auer, 2002] Peter Auer. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3(Nov):397–422, 2002.

[Chen *et al.*, 2013] Wei Chen, Yajun Wang, and Yang Yuan. Combinatorial multi-armed bandit: General framework, results and applications. In *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, ICML'13, pages I–151–I–159. JMLR.org, 2013.

[Chu *et al.*, 2011] Wei Chu, Lihong Li, Lev Reyzin, and Robert Schapire. Contextual bandits with linear payoff functions. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 208–214, 2011.

[Gai *et al.*, 2012] Yi Gai, Bhaskar Krishnamachari, and Rahul Jain. Combinatorial network optimization with unknown variables: Multi-armed bandits with linear rewards and individual observations. *IEEE/ACM Trans. Netw.*, 20(5):1466–1478, October 2012.

[Ghosh *et al.*, 2017] Avishek Ghosh, Sayak Ray Chowdhury, and Aditya Gopalan. Misspecified linear bandits. In *AAAI*, pages 3761–3767, 2017.

[Gopalan *et al.*, 2014] Aditya Gopalan, Shie Mannor, and Yishay Mansour. Thompson sampling for complex online problems. In *International Conference on Machine Learning*, pages 100–108, 2014.

[Greenewald *et al.*, 2017] Kristjan Greenewald, Ambuj Tewari, Susan Murphy, and Predag Klasnja. Action centered contextual bandits. In *Advances in neural information processing systems*, pages 5977–5985, 2017.

[Hoffman *et al.*, 2011] Matthew D Hoffman, Eric Brochu, and Nando de Freitas. Portfolio allocation for bayesian optimization. In *UAI*, pages 327–336. Citeseer, 2011.

[Krishnamurthy *et al.*, 2018] Akshay Krishnamurthy, Zhiwei Steven Wu, and Vasilis Syrgkanis. Semiparametric contextual bandits. *arXiv preprint arXiv:1803.04204*, 2018.

[Kveton *et al.*, 2015a] Branislav Kveton, Csaba Szepesvári, Zheng Wen, and Azin Ashkan. Cascading bandits: Learning to rank in the cascade model. In *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, pages 767–776. JMLR.org, 2015.

[Kveton *et al.*, 2015b] Branislav Kveton, Zheng Wen, Azin Ashkan, and Csaba Szepesvári. Combinatorial cascading bandits. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, pages 1450–1458, Cambridge, MA, USA, 2015. MIT Press.

[Li *et al.*, 2010] Lihong Li, Wei Chu, John Langford, and Robert E. Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 661–670, New York, NY, USA, 2010. ACM.

[Li *et al.*, 2016] Shuai Li, Baoxiang Wang, Shengyu Zhang, and Wei Chen. Contextual combinatorial cascading bandits. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, pages 1245–1253. JMLR.org, 2016.

[Ou *et al.*, 2019] Mingdong Ou, Nan Li, Cheng Yang, Shenghuo Zhu, and Rong Jin. Semi-parametric sampling for stochastic bandits with many arms. In *Proc. of AAAI Conference on Artificial Intelligence*, Hawaii USA, Jan 2019.

[Press, 2009] William H Press. Bandit solutions provide unified ethical models for randomized clinical trials and comparative effectiveness research. *Proceedings of the National Academy of Sciences*, pages pnas–0912378106, 2009.

[Qin *et al.*, 2014] Lijing Qin, Shouyuan Chen, and Xiaoyan Zhu. Contextual combinatorial bandit and its application on diversified online recommendation. In Mohammed Javeed Zaki, Zoran Obradovic, Pang-Ning Tan, Arindam Banerjee, Chandrika Kamath, and Srinivasan Parthasarathy, editors, *Proceedings of the 2014 SIAM International Conference on Data Mining*, pages 461–469. SIAM, 2014.

[Robbins, 1952] Herbert Robbins. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5):527–535, 1952.

[Schwartz *et al.*, 2017] Eric M Schwartz, Eric T Bradlow, and Peter S Fader. Customer acquisition via display advertising using multi-armed bandit experiments. *Marketing Science*, 36(4):500–522, 2017.

[Thompson, 1933] William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

[Xie *et al.*, 2015] Miao Xie, Qiusong Yang, Qing Wang, Gao Cong, and Gerard De Melo. Dynadiffuse: A dynamic diffusion model for continuous time constrained influence maximization. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.