Federated Bandit: A Gossiping Approach

Zhaowei Zhu[*†], **Jingxuan Zhu**[*§], **Ji Liu**[§], **and Yang Liu**[†]
[†]UC Santa Cruz, [§]Stony Brook University[*]
zwzhu@ucsc.edu, {jingxuan.zhu,ji.liu}@stonybrook.edu, yangliu@ucsc.edu

Abstract

In this paper, we study *Federated Bandit*, a decentralized Multi-Armed Bandit problem with a set of N agents, who can only communicate their local data with neighbors described by a connected graph G. Each agent makes a sequence of decisions on selecting an arm from M candidates, yet they only have access to local and potentially biased feedback/evaluation of the true reward for each action taken. Learning only locally will lead agents to sub-optimal actions while converging to a no-regret strategy requires a collection of distributed data. Motivated by the proposal of federated learning, we aim for a solution with which agents will never share their local observations with a central entity, and will be allowed to only share a private copy of his/her own information with their neighbors. We first propose a decentralized bandit algorithm `Gossip_UCB`, which is a coupling of variants of both the classical gossiping algorithm and the celebrated Upper Confidence Bound (UCB) bandit algorithm. We show that `Gossip_UCB` successfully adapts local bandit learning into a global gossiping process for sharing information among connected agents, and achieves guaranteed regret at the order of $O(\max\{\texttt{poly}(N,M)\log T, \texttt{poly}(N,M)\log_{\lambda_2^{-1}} N\})$ for all N agents, where $\lambda_2 \in (0,1)$ is the second largest eigenvalue of the expected gossip matrix, which is a function of G. We then propose `Fed_UCB`, a differentially private version of `Gossip_UCB`, where agents preserve ϵ-differential privacy of their local data while achieving $O(\max\{\frac{\texttt{poly}(N,M)}{\epsilon}\log^{2.5} T, \texttt{poly}(N,M)(\log_{\lambda_2^{-1}} N + \log T)\})$ regret.

1 Introduction

When data resides at distributed ends, soliciting them to a single server to perform centralized learning might compromise users' privacy. Among all solutions, federated learning (FL) [1, 2] arises as a promising paradigm, where massive users are allowed to collaboratively train a model while keeping the training data decentralized at local. In this paper, we introduce `federated bandit` with fully decentralized users/decision-makers and heterogeneous rewards. Our aim is to provide a solution to enable collaborative learning among decentralized sequential decision-makers in the classical multi-armed bandit (MAB) setting, but with strong (i) regret guarantee even with heterogeneous reward observations, and (ii) privacy guarantees of each user's local data.

Specifically, consider the following running example. Suppose that multiple hospitals decide to test the effectiveness of different treatment plans (*arms*). Due to the limitations such as data size, health condition and demographics of the patient population, and the details of how a treatment is used [3], each individual hospital may not be able to fully and truthfully observe the effect of treatments. In other words, individual hospitals will only observe locally biased feedback on the deployed treatment (*heterogeneous rewards*). Sharing observations across institutes is necessary for the decision-making process. However, due to privacy regulation, it is hard and expensive to call for centralized efforts to coordinate a transportation of data among hospitals. On the other hand, it is relatively easier for individual hospitals to reach agreements to share their observed treatment plan and effects with several others in an ad hoc way. The hospital treatment selection problem mentioned above is effectively

[*]Equal contributions.

NeurIPS-20 Workshop on Scalability, Privacy, and Security in Federated Learning, Vancouver, Canada.

a sequential decision-making problem which can be abstracted as a MAB one. Formally, there is a group of N decision-makers facing a common set of arms. At each step $t = 1, 2, \ldots, T$, each decision-maker selects one arm in parallel. Decision-makers only have access to local *biased* rewards. Therefore, the agents' individually observed rewards do not fully reflect the true quality of each arm. Instead, the arms' true rewards are collectively decided by all decision-makers' local observations. In our heterogeneous setting, we consider a tractable scenario where the true quality of each treatment (arm) is the average of all hospitals' (agents') locally observed quality (in expectation). Each user aim to select the best arm via exchanging information only with their neighbors privately.

The key technical challenge of the above learning problem lies in the fully-decentralized information sharing and privacy protection with sequential observations. First, to reduce the communication overhead and privacy leakage during decentralized information sharing, we aim for a solution with only sharing the information over the adjacency matrix (graph) of agents in a gossiping way. However, classical gossiping methods [4] do not incorporate individual decision-maker's newly observed reward information. Thus, the resulting information at all other agents may not converge and reflect the true statistics of each arm adaptively, which is especially true in the fully-decentralized and heterogeneous settings. Secondly, even though the gossiping update is better than directly sharing data in terms of privacy, we still need a mechanism to ensure a specific privacy level in the worst case. We adopt the solution concept differential privacy (DP) [5–8] and extend our gossiping bandit results to a differentially private one.

In this paper, we attempt to solve the above federated bandit learning problem: (1) We introduce a novel extension of the classical MAB problem to a fully-decentralized federated learning setting with gossiping, where an individual decision-maker only has access to biased rewards, and agents have limited communication capacity and can only exchange their beliefs of rewards with neighbors; (2) We propose `Gossip_UCB` to solve the challenges for combining gossiping with bandit learning processes and develop novel proof techniques to guarantee its regret. (3) To ensure the differential privacy for each observation during the federated bandit learning process, we extend the proposed gossiping bandit algorithm to `Fed_UCB`, and prove agents preserve ϵ-differential privacy of their local data while achieving $O(\max\{\frac{\texttt{poly}(N,M)}{\epsilon}\log^{2.5}T, \texttt{poly}(N,M)\log_{\lambda_2^{-1}}N\})$ regret. `Fed_UCB` is also tested using real medical dataset [9]. (4) To the best of our knowledge, `Fed_UCB` is the first fully decentralized bandit learning framework that handles heterogeneous data sources with a privacy guarantee. The results lay the foundation to study more sophisticated and probably more practical settings (e.g., contextual bandit setting to further handle population biases at each local agent).

Most relevant to us are three lines of works:
Distributed MAB MAB problems have been studied within a multi-agent setting [10–15]. But these works mostly either do not consider a consensus reaching in cheap communication setting (gossiping), or do not target on heterogeneous rewards where agents' observations incorporate local bias. For example, instead of reaching consensus among agents, [11–15] focused on avoiding the collision in wireless communication or cognitive radio. The homogeneous rewards were assumed in [16–19]. However, the rewards in federated learning setting should be heterogeneous due to various limitations [3].
Information propagation and gossiping The idea of gossiping was originally proposed to solve the *consensus reaching* problem in distributed computation [20–23], and questions about gossiping convergence rate were studied in [4, 24, 25]. It has also been used to solve distributed problems, such as convex optimization, ranking, and voting problems; and more recently to computing machine learning related statistics. Notable examples include [26] for calculating PCA, [27, 28] for computing U-Statistics, [29, 30] for computing gradients, [31] for federated learning, and [32] for reinforcement learning.
Federated learning and privacy preserving bandit Due to the high demand for privacy protection across different sectors such as financial, medical, and government systems, federated learning is becoming a trending solution that has been widely discussed [1, 2, 9, 33, 34]. Recently, differential privacy has also been adopted in solving MAB problems while ensuring privacy [35–37], but they either consider a single agent problem or use a homogeneous reward setting. We will follow the idea of DP in our work and offer a theoretically rigorous treatment for our federated bandit problem.

2 Problem Formulation

Consider a network consisting of N agents. For ease of presentation, we label the agents from 1 through N. The agents are not aware of such a global labeling, but can differentiate between their

neighbors. The set of agents is denoted by $[N] = \{1, 2, \ldots, N\}$. All agents face a common set of M arms, denoted by $[M] = \{1, 2, \ldots, M\}$. At each discrete time $t \in \{1, 2, \ldots, T\}$, each agent i makes a decision on which arm to select from the M options; the selected arm is denoted by $a_i(t) \in [M]$. When agent i selects an arm $k \in [M]$, the agent collects a reward which is generated according to a random variable $X_k(t)$.[2] But the agent cannot observe its exact reward; instead, it observes a locally biased "noisy" copy of the reward, which is generated according to another random variable $X_{i,k}(t)$. The unobservability of $X_k(t)$ can be due to local observational bias. We assume that $\{X_k(t)\}_{t=1}^T$ and $\{X_{i,k}(t)\}_{t=1}^T$ are i.i.d. random processes. For simplicity of analysis, we also assume that all X_k's and $X_{i,k}$'s have bounded support $[0, 1]$. The relationship between $X_k(t)$ and $X_{i,k}(t)$ is as follows. Let μ_k and $\mu_{i,k}$ be the mean of $X_k(t)$ and $X_{i,k}(t)$, respectively. For each $k \in [M]$, the mean of arm k's reward equals the average[3] of the means of all agents' observed rewards, i.e., $\underline{\mu_k := \frac{1}{N}\sum_{i=1}^N \mu_{i,k}}$, which implies that the true reward can be obtained by averaging and thus cancelling out local biases. Note the heterogeneous reward can model the systematic observation bias or the bias of datasets, which is more general and meaningful than the homogeneous reward, especially in FL settings [1] where each agent's systematic observation bias makes the locally optimal solution does not correspond to the real optimal action. Although the bias of datasets is probably more suitable to be modeled as contextual bandits in practice, we currently focus on the classical bandit setting for a theoretically sound solution, which is also an essential foundation for future practically feasible extensions. Without loss of generality, suppose $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_M$, which implies that arm 1 is the best option. The difference of each arm's mean reward is denoted by $\Delta_k = \mu_1 - \mu_k$. The *federated bandit problem* is for each agent i to minimize the following (weak) *regret*: $\underline{R_i(T) = T\mu_1 - \sum_{t=1}^T \mathbb{E}\left[X_{a_i(t)}(t)\right]}$ with the goal of achieving $R_i(T) = o(T)$ (i.e., $R_i(T)/T \to 0$ as $T \to \infty$) for all $i \in [N]$. It is worth noting that each agent i only observes $X_{i,k}$, $k \in [M]$, and $\mu_{i,1}$ is *not* necessarily the largest among $\mu_{i,1}, \mu_{i,2}, \ldots, \mu_{i,M}$. A naive agent, which uses a standard centralized bandit algorithm, may not solve the problem without exchanging information with other agents.

2.1 Privacy Guarantee

The privacy of arms needs to be preserved in federated bandits. We aim to protect privacy from the source of data, i.e. the sequential observations $\{X_{i,k}(t)\}_{t=1}^T$. Differential privacy (DP) is one popular mechanism to ensure some privacy level of an algorithm $\mathcal{B}$ [5]. A DP mechanism can make the adversary hard to distinguish two adjacent streams $\{X_{i,k}(t)\}_{t=1}^T$ and $\{X'_{i,k}(t)\}_{t=1}^T$, which differ at each time t. Let $\mathcal{C}$ be the space of all possible outputs by Algorithm $\mathcal{B}$. DP is defined as:

Definition 1. *(Differential privacy [5]) A (randomized) algorithm $\mathcal{B}$ is ϵ-differentially private if for any adjacent streams $\{X_{i,k}(t)\}_{t=1}^T$ and $\{X'_{i,k}(t)\}_{t=1}^T$, and for all sets $\mathcal{O} \in \mathcal{C}$,*

$$\mathbb{P}\left[\mathcal{A}(\{X_{i,k}(t)\}_{t=1}^T) \in \mathcal{O}\right] \leq e^\epsilon \cdot \mathbb{P}\left[\mathcal{A}(\{X'_{i,k}(t)\}_{t=1}^T) \in \mathcal{O}\right].$$

2.2 Communication Graph

The neighbor relationships among the agents is described by a simple, undirected, connected graph $G = (V, E)$, whose vertices correspond to agents and whose edges depict neighbor relationships. Denote $\mathcal{N}_i$ as the set of nodes that are directly connected to agent i. Then, $\mathcal{N}_i$ is also the set of agent i's neighbors. We follow the setting in the classical gossiping [4] that at each time t, exactly one pair of two neighboring agents on an edge in E are activated and exchange information.

3 Gossip UCB

Before we offer the privacy-preserving solution, we first introduce an extension of the classical Upper Confidence Bound algorithm to a gossiping setting. As may be noticed, in the classical gossiping setting [4], the consensus is defined over initial data only. While in our setting, not only is the gossiping process required to incorporate with newly arrived data from each agent, but also the gossiped information will affect the arm to be selected and thus the observed data of an agent in the future. We present an algorithm, called `Gossip_UCB`, to solve the gossiping bandit problem. The algorithm hinges on combining and extending the classical gossiping algorithm and the celebrated

[2] Different agents may select the same arm k at same time t. If this is the case, their rewards can be different as they may collect different realizations of $X_k(t)$.

[3] It can be generalized to the cases where the global reward is defined as any "convex combination" of all local rewards following the "push sum" idea [38].

UCB1 index policy [39]; yet our algorithm requires substantial changes for both the gossiping and bandit learning steps. Several crucial steps are outlined as follows.
(1) Different from the classical gossiping algorithm, the gossiping procedure will incorporate new information from each agent's local sampling and observations at each step t. We adopt the classical gossip algorithm by adding "gradient" information at each step.
(2) Compared to standard bandit learning with only one decision maker where the traditional sample complexity bound can be employed, we need to cope with the uncertainties during gossiping.
(3) A fully-decentralized structure requires designing a local information sharing mechanism. Besides, the delayed impact of local information sharing should be bounded analytically for computing the confidence bound locally.

3.1 Preliminaries

We define several quantities that will help us present our algorithm and analysis smoothly.

Sample counts: Each agent i maintains two sets of counters:
• $n_{i,k}(t)$: the number of times agent i has sampled arm k by time t;
• $\tilde{n}_{i,k}(t)$: agent i's local estimate of global maximum of pulls on arm k and is defined as

$$\tilde{n}_{i,k}(t+1) = \max\{n_{i,k}(t), \tilde{n}_{j,k}(t), j \in \mathcal{N}_i\}. \tag{1}$$

We assume agent i can observe $\tilde{n}_{j,k}(t),\ j \in \mathcal{N}_i$, thus is able to update $\tilde{n}_{i,k}$ at each time.

Sample mean: Let $\mathbb{1}(\cdot)$ be an indicator function that returns 1 when the specific condition holds and 0 otherwise. Sample mean $\tilde{X}_{i,k}(t)$ is the average observation of agent i on arm k at time t:

$$\tilde{X}_{i,k}(t) = \frac{1}{n_{i,k}(t)} \sum_{\tau=1}^{t} \mathbb{1}(a_i(\tau) = k) \cdot X_{i,a_i(\tau)}(\tau). \tag{2}$$

Estimate of rewards: Each agent i maintains an estimate of the reward of arm k at time t, which is supposed to be unbiased and denoted by $\vartheta_{i,k}(t)$. The agents' goal is to narrow the gap between $\vartheta_{i,k}(t)$ and μ_k with sequential observations and gossiping.

Upper confidence bound: In the UCB algorithm, agent i's belief on each arm k relies on two terms: the estimate $\vartheta_{i,k}(t)$ and the upper confidence bound $C_{i,k}(t)$. The latter term denotes the uncertainty of belief. The arm to be pulled is selected as $a_i(t) = \arg\max_k\ \vartheta_{i,k}(t-1) + C_{i,k}(t)$.

Gossiping matrix: Denote by $W := \frac{1}{|E|} \sum_{(i,j)\in E} \left(I_N - \frac{1}{2}(e_i - e_j)(e_i - e_j)^\top\right)$ the gossiping matrix over G, which is a positive semi-definite matrix whose largest eigenvalue equals 1, and its second largest eigenvalue is denoted by $\lambda_2(W)$ and short-handed as λ_2 without ambiguity. Note $\lambda_2 < 1$ whenever G is connected [4].

3.2 Algorithm

`Gossip_UCB` is detailed in Algorithm 1. Each agent runs this algorithm in parallel. Note all the information is shared in a fully distributed fashion and the bandit estimate is updated in a gossiping way. There are two points worth noting.

Local information sharing Throughout the algorithm, agents need to share two local variables with their neighbors: the number of observations $n_{i,k}(t)$ and the estimate $\vartheta_{i,k}(t)$. The sample count $n_{i,k}(t)$ is shared to keep all the agents "in the same page". Note the bottleneck of a bandit problem is insufficient observations of a particular arm k, and the essential of UCB algorithms is encouraging the exploration of these "undersampled" arms. In the multi-agent scenario, we can take the advantage of neighboring agents and require some local consistency in sample counts. Particularly, we want to keep all agents' knowledge of arm k "at the same page" by encouraging $n_{i,k}(t) \geq \tilde{n}_{i,k}(t) - N$ in line 6. Recall that $\tilde{n}_{i,k}(t)$ is agent i's local estimate of global maximum number of pulls, and is updated by local observations only as is defined in the previous section. Moreover, we will prove that, this requirement helps us get rid of relying on any global sample count so that $C_{i,k}(t)$ can be computed by each agent locally. The estimate $\vartheta_{i,k}(t)$ is updated in a gossiping way.

Gossip bandit update The gossip updates are defined in line 20 and line 22. In traditional bandit problems, it is enough for each agent to maintain $\tilde{X}_{i,k}(t)$. However, in our concerned gossiping setting, solely relying on $\tilde{X}_{i,k}(t)$ may induce a biased estimate. The gossiping mechanism follows [30], where the difference $\tilde{X}_{i,k}(t) - \tilde{X}_{i,k}(t-1)$ can be seen as a gradient. Later we will show the effectiveness of the proposed gossip bandit update.

Algorithm 1: `Gossip_UCB`

Input: $G, T, C_{i,k}(t)$

1 **Initialization:** Each agent pulls each arm once, and receives a reward $X_{i,k}(0)$, $i \in [N]$, $k \in [M]$. Set $n_{i,k}(0) = 1$, $\vartheta_{i,k}(0) = \tilde{X}_{i,k}(0) = X_{i,k}(0)$.

2 **for** $t = 1, \ldots, T$ **do**

3 $\quad \mathcal{A}_i = \varnothing$

4 $\quad n_{i,k}(t) = n_{i,k}(t-1), \forall k \in [M]$

5 $\quad \tilde{n}_{i,k}(t+1) = \max\{n_{i,k}(t), \tilde{n}_{j,k}(t), j \in \mathcal{N}_i\}, \forall k \in [M]$

6 $\quad$ Put k into set $\mathcal{A}_i$ **if** $n_{i,k}(t) < \tilde{n}_{i,k}(t) - N, \forall k \in [M]$ *// local consistency requirements*

7 $\quad$ **if** $\mathcal{A}_i$ *is empty* **then**

8 $\quad\quad$ **for** $k = 1, \ldots, M$ **do**

9 $\quad\quad\quad Q_{i,k}(t) := \vartheta_{i,k}(t-1) + C_{i,k}(t)$ *// update the belief on each arm*

10 $\quad\quad\quad a_i(t) = \arg\max_k Q_{i,k}(t)$ *// select the best arm to pull*

11 $\quad\quad$ **end**

12 $\quad$ **else**

13 $\quad\quad a_i(t)$ is randomly selected from $\mathcal{A}_i$

14 $\quad$ **end**

15 $\quad$ Observe arm $a_i(t)$, get $X_{i,a_i(t)}(t)$, and update $\tilde{X}_{i,k}(t), \forall k$, following (2)

16 $\quad n_{i,a_i(t)} := n_{i,a_i(t)} + 1$

17 $\quad$ **if** *agent i is selected to gossip with agent j* **then**

18 $\quad\quad$ agent i sends $\vartheta_{i,k}(t-1)$ to agent j

19 $\quad\quad$ agent i receives $\vartheta_{j,k}(t-1)$ from agent j

20 $\quad\quad \vartheta_{i,k}(t) := \frac{\vartheta_{i,k}(t-1)+\vartheta_{j,k}(t-1)}{2} + \tilde{X}_{i,k}(t) - \tilde{X}_{i,k}(t-1)$ *// gossiping update*

21 $\quad$ **else**

22 $\quad\quad \vartheta_{i,k}(t) := \vartheta_{i,k}(t-1) + \tilde{X}_{i,k}(t) - \tilde{X}_{i,k}(t-1)$ *// normal update*

23 $\quad$ **end**

24 **end**

3.3 Theoretical Analysis

Note the arm selection in Algorithm 1 relies on the upper confidence bound $C_{i,k}(t)$. In this section, we would like to find an appropriate choice of $C_{i,k}(t)$ and derive the corresponding upper bound of each agent's regret by implementing `Gossip_UCB`.

Technical challenges The main technical challenge is tackling the *coupling effects of gossiping and bandit learning*. On a high level, classical technical results in gossiping assumed a *static* piece of information that would not change much during the entire gossiping phase. The literature [40] often adopted a phased-based learning strategy (by caching the gossiped information) to avoid changes, which effectively delays the update of learned policy (the learning needs to wait for the gossiping to converge). Since agents only share information with their neighbors, globally, there is latency in receiving this data at the non-directly connected agents. Additionally, with the existence of multiple agents, ensuring local consistency as lines 6 and 13 will incur extra delay impacts when $|\mathcal{A}_i| > 1$. The *delayed impact* affects the immediate decisions taken by other agents and further affects the gossiping process in the near future. From the bandit learning's perspective, this delay might lead to inaccurate computation of the index policies. A poorly made decision will further have cascading effects to other receiving agents, which is especially challenging in heterogeneous settings. The key step to tackle this challenge is to firstly characterize the *delayed impact* and then find an upper bound of the optimal *variance proxy* of $\vartheta_{i,k}(t)$. We have the following main result:[4]

Theorem 1. *(Main Result for `Gossip_UCB`) For the `Gossip_UCB` algorithm with bounded reward over* $[0,1]$, *and* $C_{i,k}(t) = \sqrt{\frac{2N}{n_{i,k}(t)} \log t} + \alpha_1$, *the regret of each agent i until time T satisfies*

$$R_i(t) < \sum_{\Delta_k > 0} \Delta_k \left(\max\left\{ \frac{2N}{(\frac{1}{2}\Delta_k - \alpha_1)^2} \log T, L, (3M+1)N \right\} + \alpha_2 \right).$$

where $\alpha_1 = \frac{64}{N^{17}}, \alpha_2 = (3M-1)N + \frac{2\pi^2}{3} + \frac{2\lambda_2^{1/12}}{(1-\lambda_2^{1/3})(1-\lambda_2^{1/12})}$.

We have Remark 1 for the order of regret $R_i(T)$.

[4] Full version: `https://arxiv.org/pdf/2010.12763.pdf`.

Remark 1. *There are two important terms affecting the order of regret:* $\frac{2N}{(\frac{1}{2}\Delta_k-\alpha_1)^2}\log T$ *and* L*. The order of the former term is* $O(N\log T)$*, and the order of the latter term does not depend on* T *since* L *is determined by* λ_2 *and* N*. Recall that* L *is the value which makes* $\lambda_2^{t/6}/(1-\lambda_2^{1/3}) < (Nt)^{-1}$ *hold for all* $t \geq L$*. Let* $t = 6\gamma\log_{\lambda_2^{-1}} N$*. The inequality* $\lambda_2^{t/6}/(1-\lambda_2^{1/3}) < (Nt)^{-1}$ *becomes:* $\frac{6N}{1-\lambda_2^{1/3}}\frac{1}{N^\gamma} < \frac{1}{\log_{\lambda_2^{-1}} N^\gamma}$*. There always exists a positive* γ *such that the above inequality holds. Thus the order of* $R_i(T)$ *is* $O(\max\{NM\log T, M\log_{\lambda_2^{-1}} N\})$.

As for the lower bound of the regret, consider a trivial case where the graph G is fully connected. Then easily we can show the regret of our algorithm will be lower bounded by an ideal setting where all agents will receive the reward information from everyone else simultaneously with no delay. This setting reduces to a centralized bandit setting and by calling the classical results we know the regret is lower bounded by $\Omega(\log T)$. It remains a challenging and interesting question to understand the tightness of our bound in terms of the number of agents N and the graph G.

4 `Fed_UCB`: Privacy Preserving `Gossip_UCB`

Noting directly leaking some information that might appear to be "anonymized" can be used to cross-reference with other datasets to breach privacy [41], we seek for a solution with worst-case privacy guarantees (even with arbitrarily power adversary). When guaranteeing an ϵ-differential privacy in one-shot, adding Laplacian noise $\gamma \sim \mathsf{Lap}(\frac{1}{\epsilon})$ to the observation often suffices, where a larger ϵ indicate a lower privacy level. However, preserving privacy in a sequential setting is in general hard due to the continual and sequential revelation of observations. That is, in addition to preserving the privacy of $X_{i,k}(t)$, we also need to protect it in each $\tau = t, t+1, ..., T$ steps.

To preserve at least ϵ-DP in T time slots, a naive extension of the Laplace mechanism [5] is adding Laplacian noise $\mathsf{Lap}(\frac{T}{\epsilon})$ to each observation $X_{i,k}(t)$. The noise introduced in each time step grows linearly w.r.t. T, i.e. $O(\frac{T}{\epsilon})$. To add a mild noise and maintain the same privacy level at the same time, we apply the partial sums idea [42] to $\tilde{X}_{i,k}(t)$. Since both the gossiping information $\theta_{i,k}(t)$ and the selection information $n_{i,k}(t)$ are functions of $X_{i,k}(t)$, this approach also preserves the privacy for `Gossip_UCB` by data processing inequality. Independent Laplacian noise $\gamma \sim \mathsf{Lap}(1/\epsilon')$ is added to each partial sum if there exists observations in that partial sum. Thus the total privacy guarantee is given by $\lceil\log t\rceil\epsilon$. Set $\epsilon' := \epsilon\frac{1}{\lceil\log T\rceil}$, where ϵ is a pre-set privacy level we want to achieve. Then we will achieve at least a total $\epsilon\frac{1}{\lceil\log T\rceil}\lceil\log t\rceil \leq \epsilon$ differential privacy. We leave the detailed algorithms in the full version. With additional Laplacian noise, Theorem 1 can be extended for `Fed_UCB` as:

Theorem 2. *(Main Result for* `Fed_UCB`*) For the* ϵ*-differentially private* `Fed_UCB` *algorithm with bounded reward over* $[0,1]$*, and* $\tilde{C}_{i,k}(t) = \alpha_1 + \sqrt{2N\left(\frac{128N\log^2 T\cdot\log t\cdot\log n_{i,k}(t)}{n_{i,k}^2(t)\epsilon^2} + \frac{1}{n_{i,k}(t)}\right)\log t}$*, the regret of each agent* i *until time* T *satisfies the bound*

$$R_i(T) < \sum_{\Delta_k>0}\Delta_k\left(\max\left\{\frac{N\log T\cdot\left(1+\sqrt{1+\left(16(\frac{\Delta_k}{2}-\alpha_1)/\epsilon\right)^2\log^3 T}\right)}{(\frac{\Delta_k}{2}-\alpha_1)^2}, L, (3M+1)N\right\} + 4N\log T + \alpha_3\right),$$

where $\alpha_3 := (3M-1)N + \frac{2\pi^2}{3} + \frac{2\lambda_2^{1/12}}{(1-\lambda_2^{1/3})(1-\lambda_2^{1/12})} + 4N$.

Remark 2. *The order of* $R_i(T)$ *is* $O(\max\{\frac{NM}{\epsilon}\log^{2.5} T, M(N\log T + \log_{\lambda_2^{-1}} N)\})$.

5 Conclusion

We have proposed `Gossip_UCB` for solving a gossiping bandit learning problem, where a network of agents aim to learn to converge to selecting the best arm both locally and globally through gossiping, and its differentially private variant, `Fed_UCB`, for preserving ϵ-differential privacy of the agents' local data. We have shown both `Gossip_UCB` and `Fed_UCB` achieve weak regret at an order depending on the size of agents, the number of arms, time horizon, and connectivity of the graph. Proofs, algorithms, and numerical results are available in the full version.

References

[1] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–19, 2019.

[2] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Keith Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *arXiv preprint arXiv:1912.04977*, 2019.

[3] Qinbin Li, Zeyi Wen, and Bingsheng He. Federated learning systems: Vision, hype and reality for data privacy and protection. *arXiv preprint arXiv:1907.09693*, 2019.

[4] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. Randomized gossip algorithms. *IEEE Transactions on Information Theory*, 52(6):2508–2530, 2006.

[5] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284. Springer, 2006.

[6] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211–407, 2014.

[7] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*, pages 1054–1067, 2014.

[8] Bolin Ding, Janardhan Kulkarni, and Sergey Yekhanin. Collecting telemetry data privately. In *Advances in Neural Information Processing Systems*, pages 3571–3580, 2017.

[9] Beata Strack, Jonathan P DeShazo, Chris Gennings, Juan L Olmo, Sebastian Ventura, Krzysztof J Cios, and John N Clore. Impact of hba1c measurement on hospital readmission rates: analysis of 70,000 clinical database patient records. *BioMed research international*, 2014, 2014.

[10] P. Landgren, V. Srivastava, and N. E. Leonard. Distributed cooperative decision-making in multiarmed bandits: Frequentist and bayesian algorithms. In *Proceedings of the 55th IEEE Conference on Decision and Control*, pages 167–172, 2016.

[11] N. Nayyar, D. Kalathil, and R. Jain. On regret-optimal learning in decentralized multi-player multi-armed bandits. *IEEE Transactions on Control of Network Systems*, 5(1):597–606, 2016.

[12] Keqin Liu and Qing Zhao. Distributed learning in multi-armed bandit with multiple players. *IEEE Transactions on Signal Processing*, 58(11):5667–5681, 2010.

[13] Ilai Bistritz and Amir Leshem. Distributed multi-player bandits - a game of thrones approach. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 7222–7232. Curran Associates, Inc., 2018.

[14] Dileep Kalathil, Naumaan Nayyar, and Rahul Jain. Decentralized learning for multiplayer multiarmed bandits. *IEEE Transactions on Information Theory*, 60(4):2331–2345, 2014.

[15] Aristide CY Tossou and Christos Dimitrakakis. Differentially private, multi-agent multi-armed bandits. In *European Workshop on Reinforcement Learning (EWRL)*, 2015.

[16] Abishek Sankararaman, Ayalvadi Ganesh, and Sanjay Shakkottai. Social learning in multi agent multi armed bandits. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 3(3):1–35, 2019.

[17] Yuanhao Wang, Jiachen Hu, Xiaoyu Chen, and Liwei Wang. Distributed bandit learning: Near-optimal regret with efficient communication. In *International Conference on Learning Representations*, 2020.

[18] Mithun Chakraborty, Kai Yee Phoebe Chua, Sanmay Das, and Brendan Juba. Coordinated versus decentralized exploration in multi-agent multi-armed bandits. In *IJCAI*, pages 164–170, 2017.

[19] David Martínez-Rubio, Varun Kanade, and Patrick Rebeschini. Decentralized cooperative stochastic bandits. In *Advances in Neural Information Processing Systems*, pages 4531–4542, 2019.

[20] L. Moreau. Stability of multi-agent systems with time-dependent communication links. *IEEE Transactions on Automatic Control*, 50(2):169–182, 2005.

[21] W. Ren and R. W. Beard. Consensus seeking in multiagent systems under dynamically changing interaction topologies. *IEEE Transactions on Automatic Control*, 50(5):655–661, 2005.

[22] A. Kashyap, T. Başar, and R. Srikant. Quantized consensus. *Automatica*, 43(7):1192–1203, 2007.

[23] B. Touri and A. Nedić. Product of random stochastic matrices. *IEEE Transactions on Automatic Control*, 59(2):437–448, 2014.

[24] M. Cao, D. A. Spielman, and A. S. Morse. A lower bound on convergence of a distributed network consencus algorithm. In *Proceedings of the 44th IEEE Conference on Decision and Control*, pages 2356–2361, 2005.

[25] A. Olshevsky and J. N. Tsitsiklis. Convergence speed in distributed consensus and averaging. *SIAM Journal on Control and Optimization*, 48(1):33–55, 2009.

[26] Satish Babu Korada, Andrea Montanari, and Sewoong Oh. Gossip PCA. In *Proceedings of the ACM SIGMETRICS joint international conference on Measurement and modeling of computer systems*, pages 209–220. ACM, 2011.

[27] Kristiaan Pelckmans and Johan AK Suykens. Gossip algorithms for computing u-statistics. *IFAC Proceedings Volumes*, 42(20):48–53, 2009.

[28] Igor Colin, Aurélien Bellet, Joseph Salmon, and Stéphan Clémençon. Extending gossip algorithms to distributed estimation of u-statistics. In *Advances in Neural Information Processing Systems*, pages 271–279, 2015.

[29] Benjamin Sirb and Xiaojing Ye. Decentralized consensus algorithm with delayed and stochastic gradients. *SIAM Journal on Optimization*, 28(2):1232–1254, 2018.

[30] Yang Liu, Ji Liu, and Tamer Basar. Differentially private gossip gradient descent. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 2777–2782. IEEE, 2018.

[31] István Hegedűs, Gábor Danner, and Márk Jelasity. Gossip learning as a decentralized alternative to federated learning. In *IFIP International Conference on Distributed Applications and Interoperable Systems*, pages 74–90. Springer, 2019.

[32] Joshua Romoff, Nicolas Ballas, Joelle Pineau, Mike Rabbat, et al. Gossip-based actor-learner architectures for deep reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 13299–13309, 2019.

[33] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. Practical secure aggregation for federated learning on user-held data. *arXiv preprint arXiv:1611.04482*, 2016.

[34] Jakub Konečný, H. Brendan McMahan, Felix X. Yu, Peter Richtarik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. In *NIPS Workshop on Private Multi-Party Machine Learning*, 2016.

[35] Nikita Mishra and Abhradeep Thakurta. Private stochastic multi-arm bandits: From theory to practice. 2014.

[36] Aristide CY Tossou and Christos Dimitrakakis. Algorithms for differentially private multi-armed bandits. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[37] Mohammad Malekzadeh, Dimitrios Athanasakis, Hamed Haddadi, and Ben Livshits. Privacy-preserving bandits. In *Proceedings of Machine Learning and Systems 2020*, pages 350–362. 2020.

[38] David Kempe, Alin Dobra, and Johannes Gehrke. Gossip-based computation of aggregate information. In *44th Annual IEEE Symposium on Foundations of Computer Science, 2003. Proceedings.*, pages 482–491. IEEE, 2003.

[39] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.

[40] Balazs Szorenyi, Róbert Busa-Fekete, István Hegedus, Róbert Ormándi, Márk Jelasity, and Balázs Kégl. Gossip-based distributed stochastic bandit algorithms. In *International Conference on Machine Learning*, pages 19–27, 2013.

[41] Latanya Sweeney. Simple demographics often identify people uniquely. *Health (San Francisco)*, 671(2000):1–34, 2000.

[42] T-H Hubert Chan, Elaine Shi, and Dawn Song. Private and continual release of statistics. *ACM Transactions on Information and System Security (TISSEC)*, 14(3):26, 2011.