Federated Bandit: A Gossiping Approach

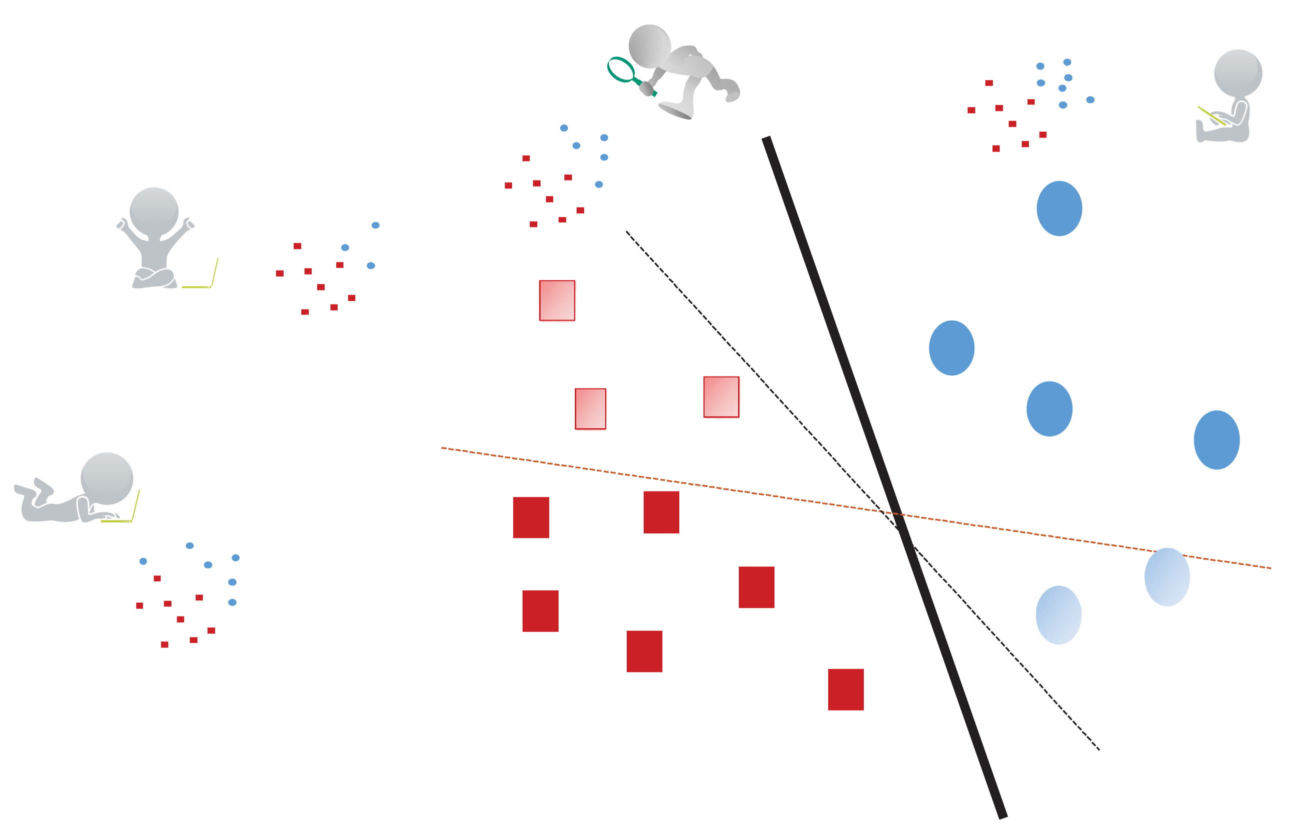
Zhaowei Zhu*†, Jingxuan Zhu*§, Ji Liu§, Yang Liu† (*Equal contributions)

†UC Santa Cruz, §Stony Brook University

{zwzhu,yangliu}@ucsc.edu, {jingxuan.zhu,ji.liu}@stonybrook.edu

Motivating Example

Story: Multiple hospitals test different treatment plans (*arms*)
Challenges: Heterogeneous feedbacks (biased decisions) & privacy concern



Problems & Solutions (Overview)

One-sentence summary:

`Fed_UCB` is a novel *fully-decentralized* bandit learning framework that handles *heterogeneous* data sources with a *privacy* guarantee.

Problems:

1. *(Decentralization)* Centralized learning requires soliciting data from distributed ends to a single server, which might compromise users' privacy
2. *(Heterogeneity)* Multiple agents may hold different and heterogeneous datasets for the same task due to the local bias
3. *(Privacy)* Directly leaking some information that might appear to be "anonymized" can be used to cross-reference with other datasets to breach privacy

Challenges:

1. *(Decentralization)* No central-controller
2. *(Heterogeneity)* Bias in local learning & problem may not be solved
3. *(Privacy)* Protect agents' privacy in the worst cases during cooperation

Solutions:

1. *(Decentralization+Heterogeneity)* `Gossip_UCB`
2. *(`Gossip_UCB`+Privacy)* `Fed_UCB`: Differentially private `Gossip_UCB`

MAB with Heterogeneous Rewards

Problem settings:

- Multi-armed bandit (MAB) with N agents and M arms;
- Each agent i selects an arm $a_i(t)$ at time t.
- $X_k(t)$ is *supposed* to be collected when arm k is pulled at time t. BUT it is unobservable.
- Actual observation: $X_{i,k}(t)$ (a locally biased "noisy" copy of $X_k(t)$);
- Relationship: $\mu_k = \mathbb{E}[X_k(t)]$, $\mu_{i,k} = \mathbb{E}[X_{i,k}(t)]$, $\mu_k := \frac{1}{N}\sum_{i=1}^N \mu_{i,k}$, $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_M$

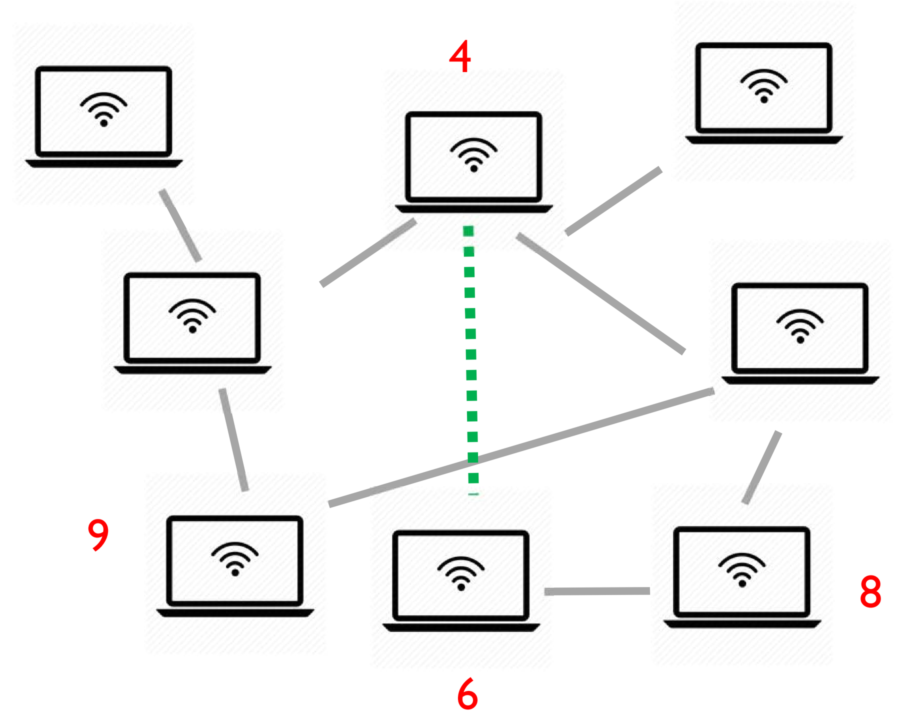
Goal: Without sharing local observations with a central entity, minimize:

$$\text{Regret:} \quad R_i(T) = T\mu_1 - \sum_{t=1}^{T} \mathbb{E}\left[X_{a_i(t)}(t)\right] .$$

Information Propagation via Gossiping

Gossiping:

- One edge activated at each t;
- Selected agents on the edge exchange information;
- Others do not update.



Key challenges:

- *Sample counts:* (agent i, arm k, time t)
  - $n_{i,k}(t)$: the number of observations (determining the quality of decision);
  - $\tilde{n}_{i,k}(t)$: local estimate of $\max_j \ n_{j,k}(t)$ (controlling the local consistency).
- *Sample mean $\tilde{X}_{i,k}(t)$ (biased)* .
- *Estimate of the average reward $\vartheta_{i,k}(t)$:* The gap $|\vartheta_{i,k}(t) - \mu_k|$ is supposed to decrease with sequential observations and gossiping.
- *Upper confidence bound (UCB):* Design the UCB $C_{i,k}(t)$ and select arm:

$$a_i(t) = \arg\max_k \ \vartheta_{i,k}(t-1) + C_{i,k}(t).$$

Algorithm (sketched):

1. *Initialization:* Each agent pulls each arm once.
2. *Local consistency check using $\tilde{n}_{i,k}(t)$ (each-t)*
3. *Locally consistent decision making (each-t):*
   - Locally consistent $\rightarrow$ $a_i(t) = \arg\max_k \ \vartheta_{i,k}(t-1) + C_{i,k}(t)$
   - Consistency violation $\rightarrow$ push local consistency
4. *Gossiping (each-t):*
   - Gossiping update: $\vartheta_{i,k}(t) := \frac{\vartheta_{i,k}(t-1)+\vartheta_{j,k}(t-1)}{2} + \tilde{X}_{i,k}(t) - \tilde{X}_{i,k}(t-1)$;
   - Normal update: $\vartheta_{i,k}(t) := \vartheta_{i,k}(t-1) + \tilde{X}_{i,k}(t) - \tilde{X}_{i,k}(t-1)$.

Regret Upper Bound for `Gossip_UCB`

Theorem 1. *(Regret upper bound for `Gossip_UCB`) For the `Gossip_UCB` algorithm with bounded reward over $[0,1]$, and*

$$C_{i,k}(t) = \sqrt{\frac{2N}{n_{i,k}(t)} \log t + \alpha_1}, \qquad (1)$$

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

Remark: The order of $R_i(T)$ is $O(\max\{NM\log T, M\log_{\lambda_2^{-1}} N\})$.

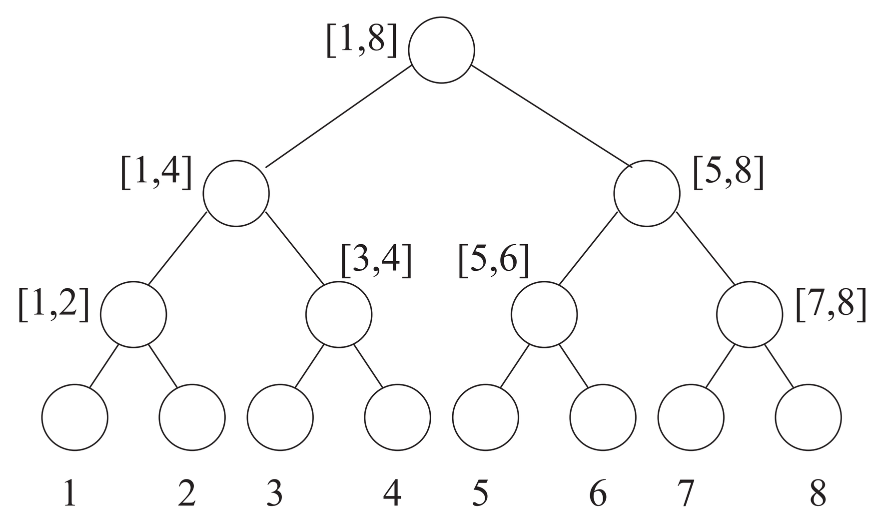
`Fed_UCB`: Privacy Preserving `Gossip_UCB`

Differential privacy (DP):

- *(Definition)* 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 $\mathcal{O} \in \mathcal{C}$,

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

- *(Online DP)* Guarantee ϵ-DP on every T.
  - *(Naive)* Adding Laplacian noise Lap($\frac{T}{\epsilon}$) to each observation $X_{i,k}(t)$ (too large)
  - *(Partial sum)* Adding Laplacian noise Lap($\frac{\lceil \log T \rceil}{\epsilon}$) following a binary tree.



Example:

Node 4: $\text{Noise}_{[1,4]}$
Node 6: $\text{Noise}_{[1,4]} + \text{Noise}_{[5,6]}$
Node 7: $\text{Noise}_{[1,4]} + \text{Noise}_{[5,6]} + \text{Noise}_7$

Remark: The order of $R_i(T)$ for ϵ-differentially private `Fed_UCB` is $O(\max\{\frac{NM}{\epsilon}\log^{2.5} T, M(N\log T + \log_{\lambda_2^{-1}} N)\})$.