Recent Theoretical Advances in Differentially Private RL

Xingyu Zhou

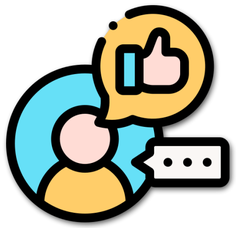
Wayne State University

Tutorials @ SIGMETRICS 2025

June 9, 2025

RL is “everywhere”…

from everyday decisions to frontier AI



Personalized Recommendation

Learn from user interactions to personalize content in ads, healthcare, and beyond

Autonomous Driving

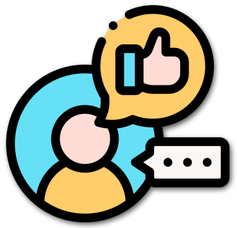
Learn from driving behavior to optimize comfort, safety, and vehicle control

LLMs Post-training

Learn from human feedback to improve model alignment and reasoning

But…Data is Private…

from medical history to preference



Personalized Recommendation

Autonomous Driving

LLMs Post-training

Learn from **user interactions** to personalize content in ads, healthcare, and beyond

Learn from **driving behavior** to optimize comfort, safety, and vehicle control

Learn from **human feedback** to improve model alignment and reasoning

"I've taken the suggested medicine for diabetes—feeling good now"

"I usually take local roads because I get anxious on highways"

"I choose the second answer because it handles my breakup more gently."

But…Data is Private…

from medical history to preference

Personalized Recommendation

How to ensure a mathematically rigorous privacy protection?

Autonomous Driving

LLMs Post-training

Learn from **user interactions** to personalize content in ads, healthcare, and beyond

Learn from **driving behavior** to optimize comfort, safety, and vehicle control

Learn from **human feedback** to improve model alignment and reasoning

Differential Privacy (DP)

The de facto mathematical framework for private data analysis—with rigorous guarantees and real-world deployment*

* Apple, Google, Meta, Microsoft, U.S. Census Bureau

Definition

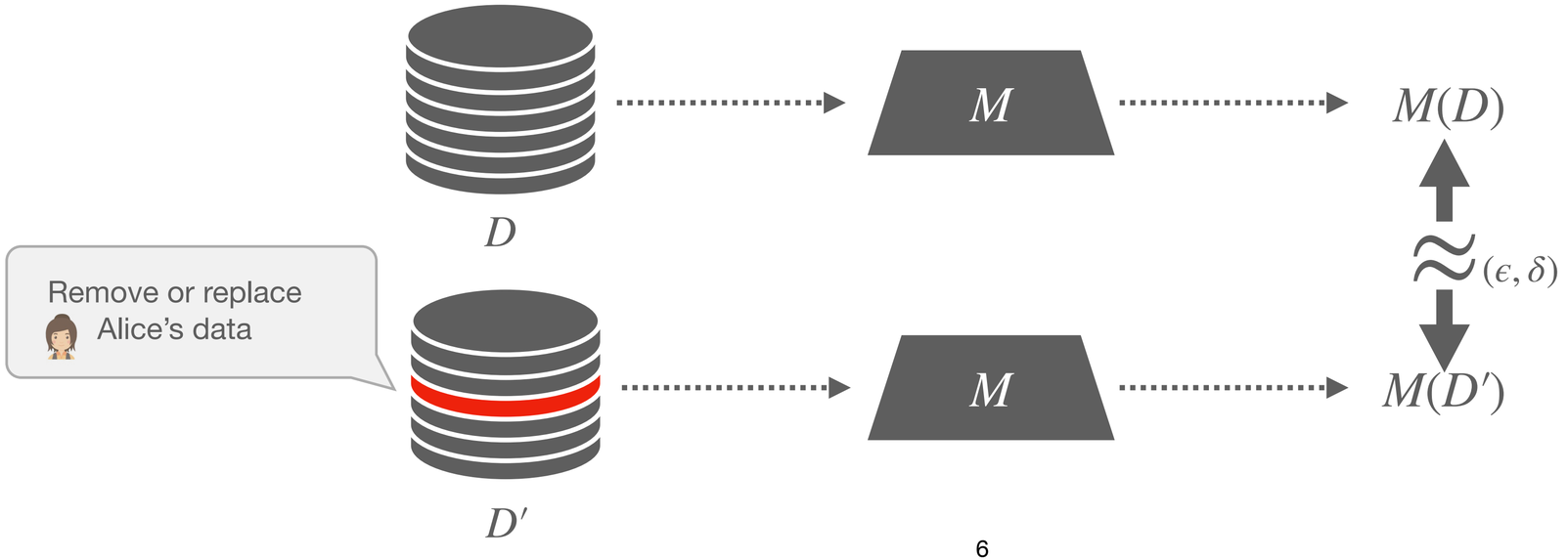
a worst-case guarantee

Definition (DP[DMNS06])

A random mechanism M is said to be (ϵ, δ)-DP if for any adjacent datasets D and D' differing in one record, any $S \subseteq \text{Range}(M)$:

$$\mathbb{P}[M(D) \in S] \leq e^{\varepsilon} \cdot \mathbb{P}[M(D') \in S] + \delta$$

If $\delta = 0$, it is *pure DP*; otherwise, *approximate DP*



Definition

a worst-case guarantee

Definition (DP[DMNS06])

A random mechanism M is said to be (ϵ, δ)-DP if for any adjacent datasets D and D' differing in one record, any $S \subseteq \text{Range}(M)$:

$$\mathbb{P}[M(D) \in S] \leq e^{\varepsilon} \cdot \mathbb{P}[M(D') \in S] + \delta$$

If $\delta = 0$, it is *pure DP*; otherwise, *approximate DP*

Remarks 📝

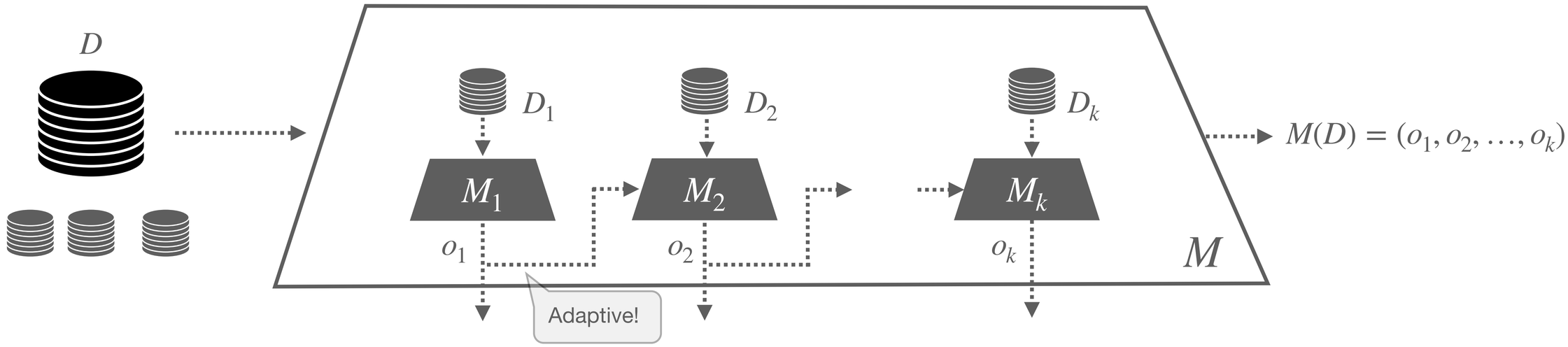
1. **Interpretation:** Bound the information gain of the adversary after observing the output
   - $\epsilon = 1.1, \delta = 0$, a prior of $50\,\% \rightarrow 75\,\%$ posterior at most
2. **Strong guarantees:** Arbitrary side information and computation power
   - adversary knows all datapoints except Alice's
3. **Various "closeness" measures:** Besides *hockey-stick divergence,* other divergences:
   - Rényi divergence gives Rényi DP[M17], closely related to zCDP [BS16]; Better composition of DP

Composition

key to the success of DP

Theorem (Parallel Composition[M09])

Suppose D is union of k disjoint datasets and each M_i is (ϵ_i, δ_i)-DP*, then M is $(\max_i \epsilon_i, \max_i \delta_i)$-DP



*Condition on all previous outputs: $o_1, \ldots, o_{i-1}$

Composition

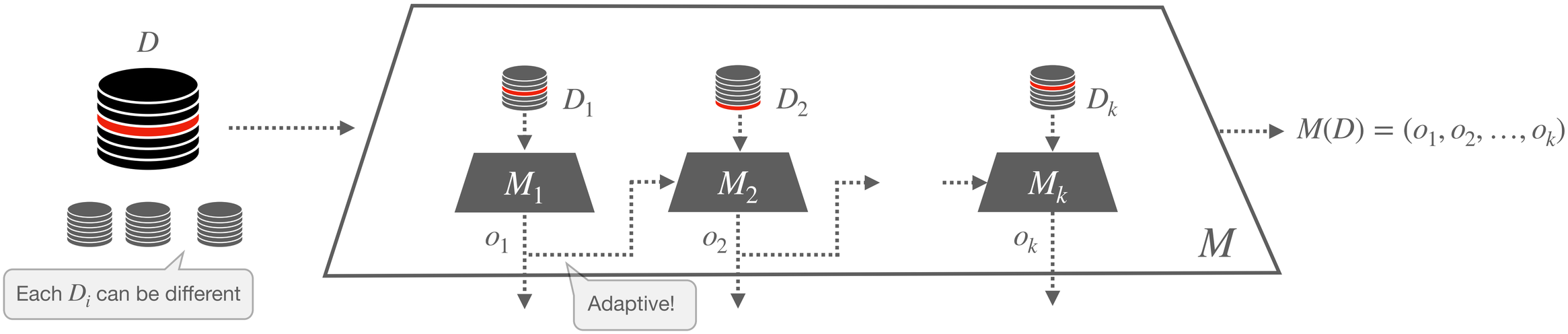
key to the success of DP

Theorem (Basic Sequential Composition[DMNS06])

Suppose $D = \cup_{i=1}^{k} D_i$ and each M_i is (ϵ_i, δ_i)-DP*, then M is $(\sum_i \epsilon_i, \sum_i \delta_i)$-DP

Implies **post-processing**
Once private, always private, if no further touch

k ϵ-DP mechanisms gives $k\epsilon$-DP



*Condition on all previous outputs: $o_1, \ldots, o_{i-1}$

Composition

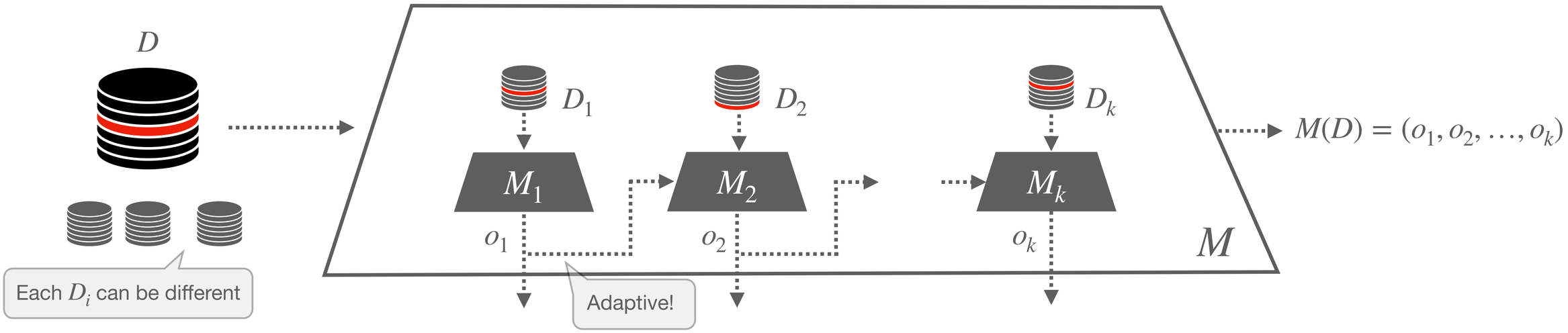
key to the success of DP

Theorem (Advanced Sequential Composition[DRV10])

Suppose $D = \cup_{i=1}^{k} D_i$ and each M_i is ϵ-DP*, then M is (ϵ', δ')-DP for any $\delta' \geq 0$ with

$$\epsilon' = \epsilon\sqrt{2k\ln(1/\delta')} + k\epsilon(e^{\epsilon} - 1)$$

For small ϵ, now $\sqrt{k}$ rather than k, if approximate DP



*Hold for approximate DP with additional $k\delta$ in δ'

Basic DP Mechanisms

from additive noise to sampling and back

DP from additive noise

Given a data analytical function $f : \mathcal{X}^n \to R^d$, the corresponding private mechanism is

$$M(D) = f(D) + Z \cdot \Delta, \text{ with } \Delta := \sup_{D \sim D'} \left\| f(D) - f(D') \right\|$$

Laplace mechanism: $Z \sim \text{Lap}(1/\epsilon)^d$ and $\|\cdot\| = \|\cdot\|_1$, then M is ϵ-DP

Gaussian mechanism: $Z \sim \mathcal{N}(0, \sigma^2)^d$, $\sigma = \sqrt{2\log(1.25/\delta)}/\epsilon$ and $\|\cdot\| = \|\cdot\|_2$, then M is (ϵ, δ)-DP*

Applications ⚙️ Private mean estimation of n bounded unit L_2-norm vectors $\{x_i\}_{i=1}^n$

$$f(D) = \frac{1}{n} \sum_i x_i$$

- **Laplace mechanism:** $\Delta = \sqrt{d}/n$, pure DP with MSE of $O(d^2/(n^2\epsilon^2))$
- **Gaussian mechanism:** $\Delta = 1/n$, approximate DP with MSE of $O(d\log(1/\delta)/(n^2\epsilon^2))$

$\approx d$ separation
Both rates are optimal, see my blog

*Holds for $\epsilon \leq 1$

Basic DP Mechanisms

from additive noise to sampling and back

DP from sampling — Exponential Mechanism [MT07]

Given a score function $q : \mathcal{X}^n \times \mathcal{H} \to R$ and D, sample an outcome $h \in \mathcal{H}$ with probability

$$\mathbb{P}(h) \propto \exp(\epsilon \cdot q(D,h)/(2\Delta)), \text{ with } \Delta := \sup_{D \sim D', h \in \mathcal{H}} |q(D,h) - q(D',h)|$$

This satisfies ϵ-DP

Remarks 📝

1. **Utility:** The sampled h satisfies that with prob. $1-\beta$

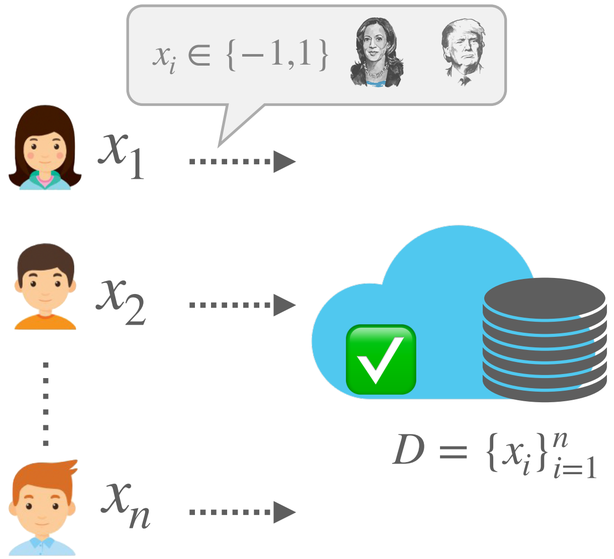
$$q(D,h) \geq \max_{h' \in \mathcal{H}} q(D,h') - \frac{2\Delta \log(|\mathcal{H}|/\beta)}{\varepsilon}$$

2. **Gumbel max trick:** finite $\mathcal{H}$, return $\arg\max_{h \in \{1,\ldots,|\mathcal{H}|\}} \left(q(D,h) + Z_h\right)$ with Gumbel noise → Exp. Mechanism

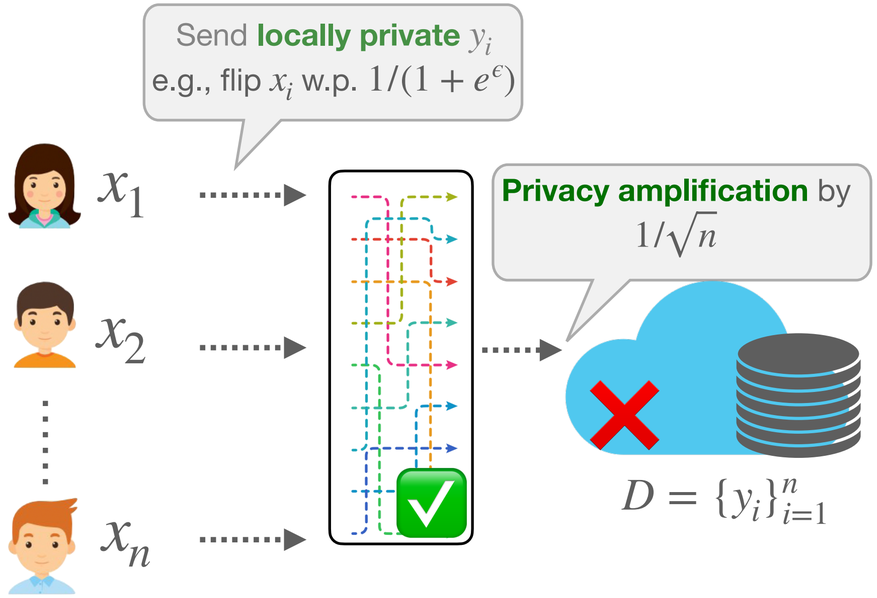
3. **Recover Laplace mechanism:** A proper choice of score function leads back to Lap. mechanism

Trust Models

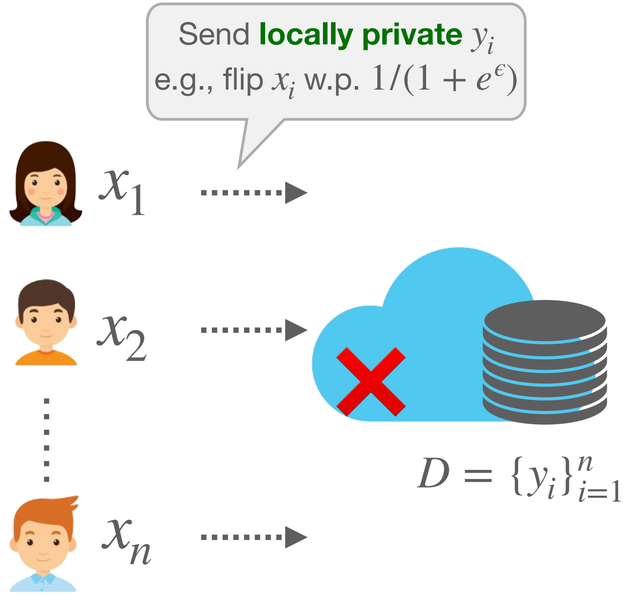
who do you trust?



Central model [DMNS06]
Server is trusted, private after server



Distributed model [CSUZZ19] (e.g., shuffle)
Third-party is trusted, private before server



Local model [KLNR08]
Server is untrusted, private after user

Summary

the DP journey so far…

1. What's DP and its guarantees

   - Limits what adversaries can learn—no matter what they already know

2. Composition

   - Privacy loss adds up—but smart composition controls the damage

3. Basic DP mechanisms

   - Add noise or sample wisely—closely related to each other

4. Three trust models

   - Who do you trust—server, third-party, or no one at all?

Summary

the DP journey so far…

1. What's DP and its guarantees

   - *Limits what a*

2. Composition

   - *Privacy loss adds up—but smart composition controls the damage*

3. Basic DP mechanisms

   - *Add noise or sample wisely—closely related to each other*

4. Three trust models

   - *Who do you trust—server, third-party, or no one at all?*

How to bridge DP with reinforcement learning?

Challenges

from privacy definition to algorithmic design

Challenge: Privacy definition

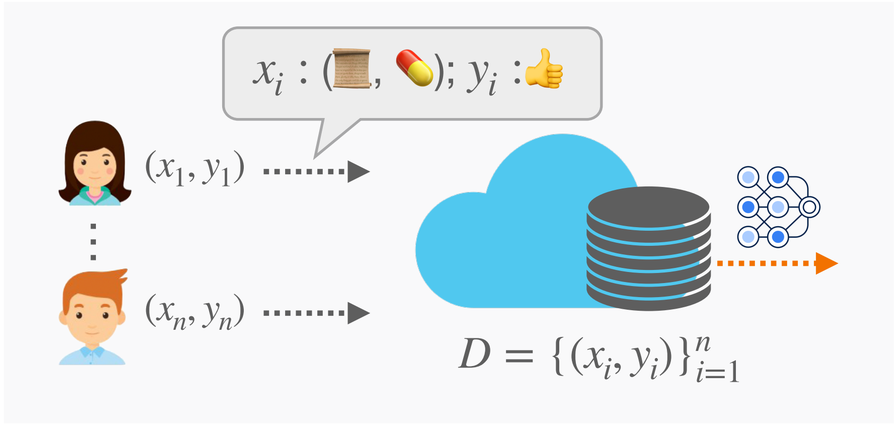
what's privacy unit and view of adv.?

Standard DP in Central Model

M is (ϵ, δ)-DP if for any D and D' differing in one record, any $S \subseteq \text{Range}(M)$:

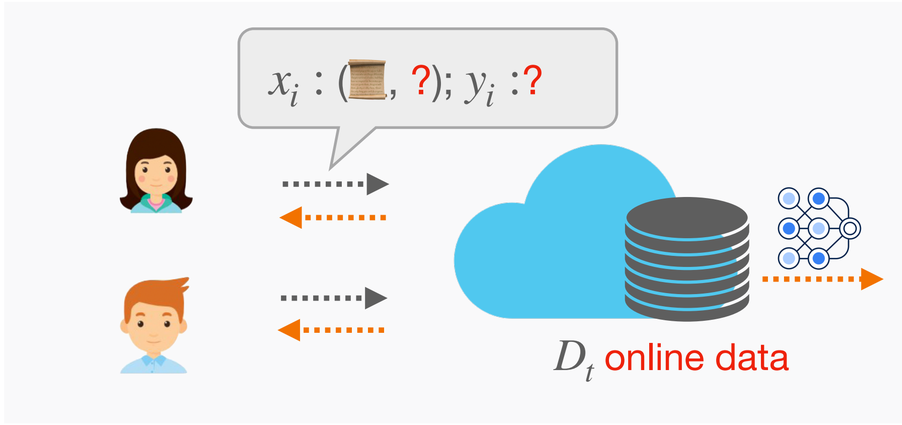
$$\mathbb{P}[M(D) \in S] \leq e^{\varepsilon} \cdot \mathbb{P}[M(D') \in S] + \delta$$

DP Supervised Learning



- **Privacy unit:** differ by one ***offline*** example
- **Adversary view:** final model

DP Reinforcement Learning

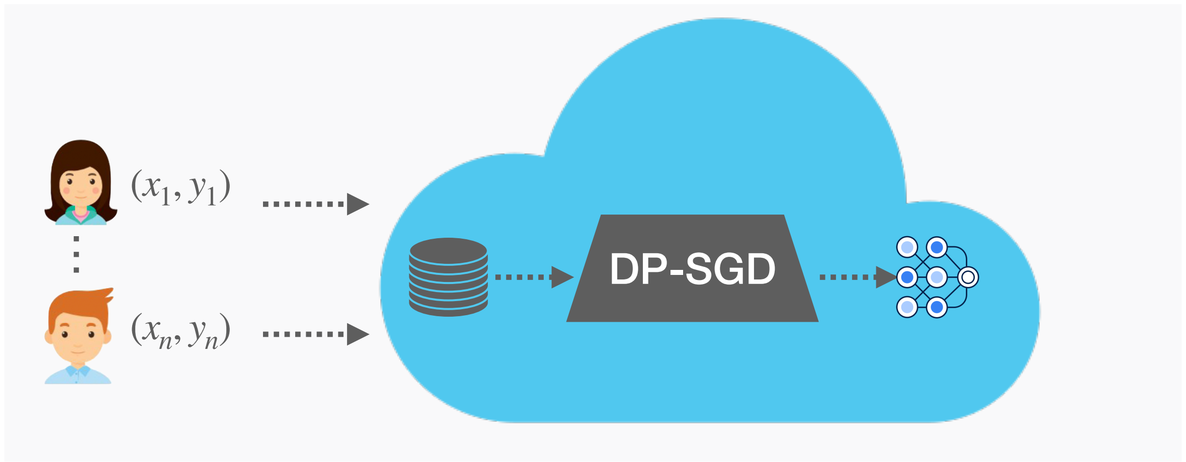


- **Privacy unit:** data point is dynamic!
- **Adversary view:** online interaction & final policy

Challenge: Algorithm design

which one to privatize and how?

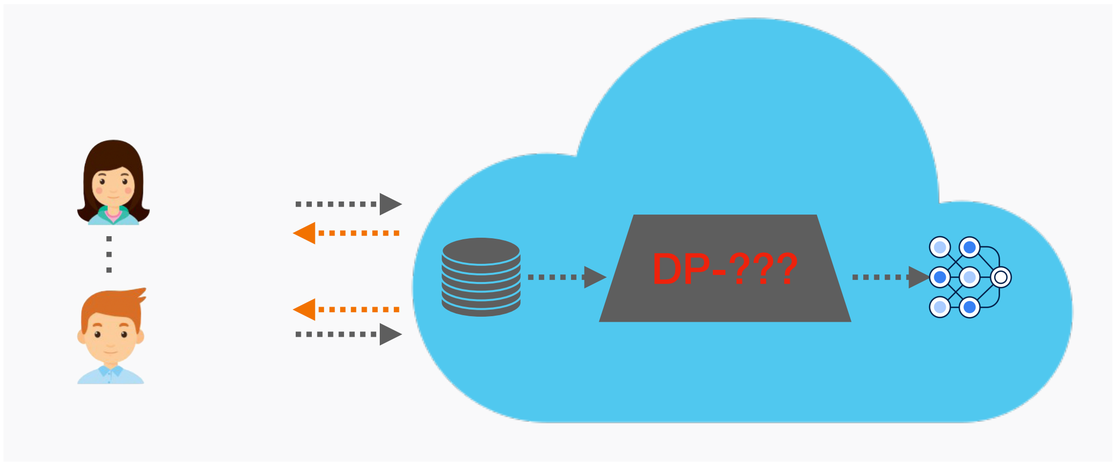
DP Supervised Learning



One universal algorithm (almost)

SGD dominates in both convex and non-convex cases

DP Reinforcement Learning



Fragmented landscape (almost*)

Different problems have different algorithms

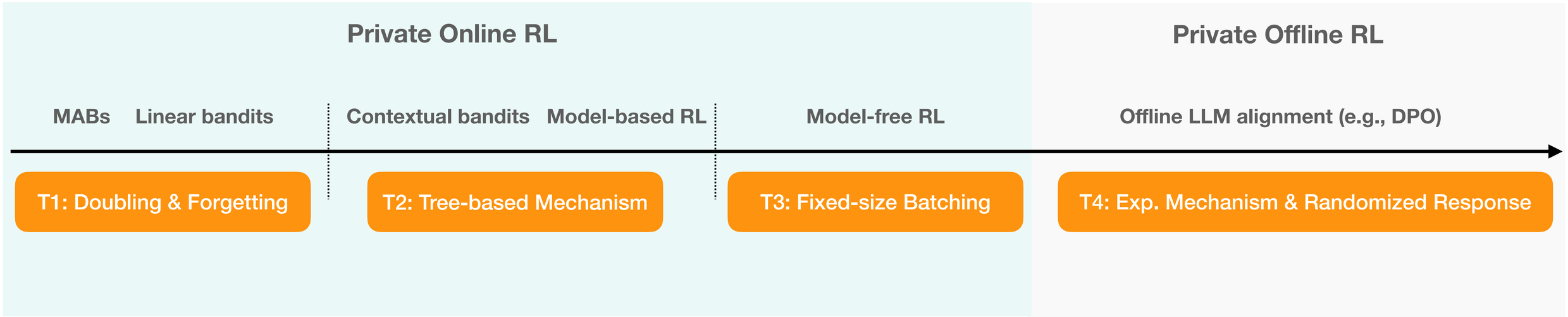
* Some recent unified framework for RL, e.g., [FKQR21]

Recent Advances

from a theoretical perspective

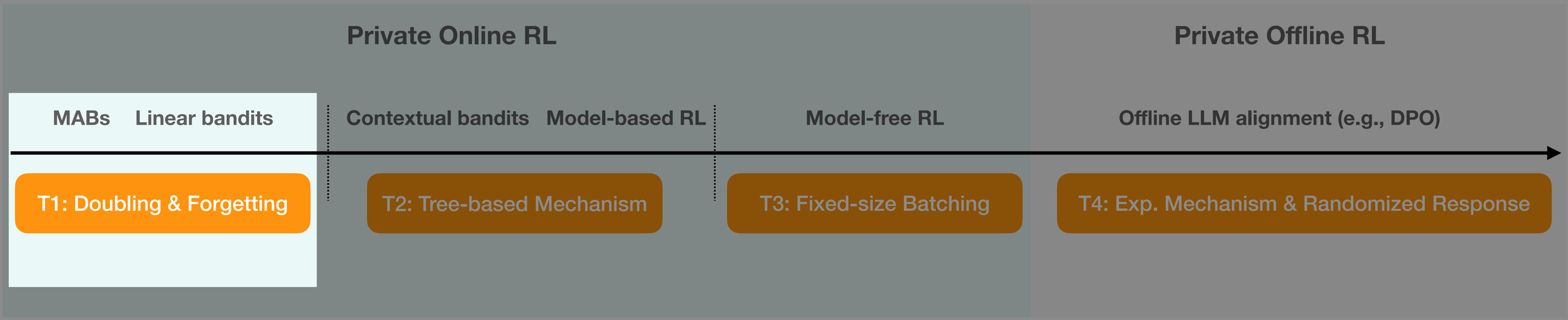
Roadmap*

4 key tools



* A (biased) selection of works :-)

Roadmap*



* A (biased) selection of works :-)

Stochastic Multi-Armed Bandit (MAB)

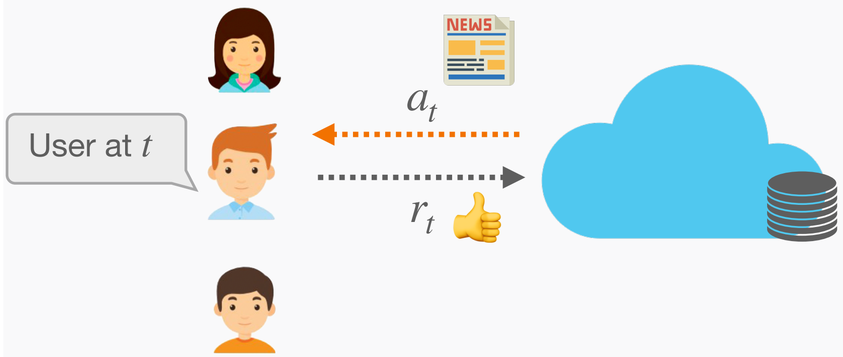
MAB setting

There are K arms. For each $t = 1, \ldots, T$:

- An arm $a_t \in [K]$ is selected
- Reward $r_t(a_t) \sim P_{a_t}$ where each P_a has support $[0,1]$ with mean μ_a (where $a^* = \arg\max_a \mu(a)$)

Goal: Minimize expected regret*

$$\mathbb{E}[R(T)] := T\mu(a^*) - \mathbb{E}[\sum r_t]$$



Successive Elimination (SE) [EDMM06]

Active set $\mathscr{A}$

Repeat till end

1. Play each arm in $\mathscr{A}$ once
2. Update $\mathscr{A}$ by removing "bad" arms

   (via count N_a and empirical estimate $\hat{\mu}_a$)

*Technically, it is pseudo-regret

DP in MAB

A common but unsatisfying def.

Definition (DP in MAB, first attempt)

An MAB algorithm M is (ϵ, δ)-DP if for all $D = (r_1, \ldots, r_T)$ and $D' = (r'_1, \ldots, r'_T)$, differing in one reward, and for all output action sequence S,

$$\mathbb{P}[M(D) \in S] \leq e^{\varepsilon} \cdot \mathbb{P}[M(D') \in S] + \delta$$

Limitations !!

1. **Not well-defined dataset**: impossible to have such a neighboring D, D'
   - Changing the reward at time t affects all future ones, due to online learning
2. **Improper privacy unit:** the true privacy target is the user at any time t
   - whether that person has participated in this process
   - what's their preference over all actions, rather than just the recommended one.

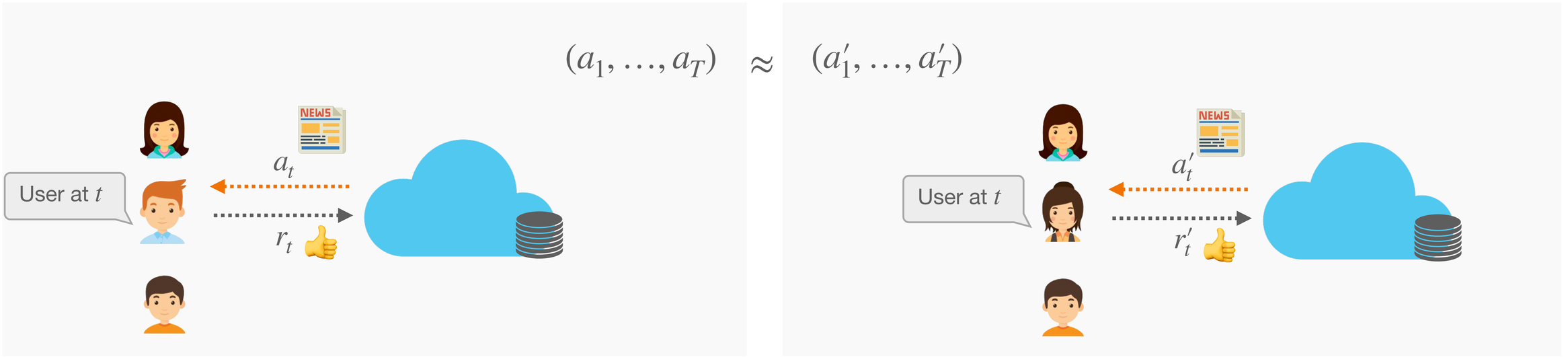
DP in MAB

A better one

Definition (DP in MAB)

An MAB algorithm M is (ϵ, δ)-DP if for all $D = (u_1, \ldots, u_T)$ and $D' = (u'_1, \ldots, u'_T)$, differing in one user, and for all output action sequence S,

$$\mathbb{P}[M(D) \in S] \leq e^{\varepsilon} \cdot \mathbb{P}[M(D') \in S] + \delta$$



See more discussion on this definition on my blog

Algorithm: DP-SE

Tool1: Doubling & Forgetting

Non-private one

Successive Elimination (SE) [EDMM06]

Active set $\mathscr{A}$

Repeat till end

1. Play each arm in $\mathscr{A}$ once
2. Update $\mathscr{A}$ by removing "bad" arms

   (via count N_a and empirical estimate $\hat{\mu}_a$)

(total rewards) / (total counts)
i.e., accumulated statistics

Private one

DP-Successive Elimination (SE) [SS19][CZ23]

Active set $\mathscr{A}$

Repeat till end

1. Play each arm in $\mathscr{A}$ for a doubling # times

   (i.e., 2^l for each batch $l = 1,2,\ldots$)
2. Update $\mathscr{A}$ by removing "bad" arms

   (via count N_a and empirical estimate $\hat{\mu}_a$ using data only in the latest batch + Laplace noise, i.e., forgetting)

Theoretical Guarantees

MAB

Theorem (DP-SE) [SS19][CZ*23]

DP-SE satisfies (ϵ, δ)-DP and achieves the following regret bound

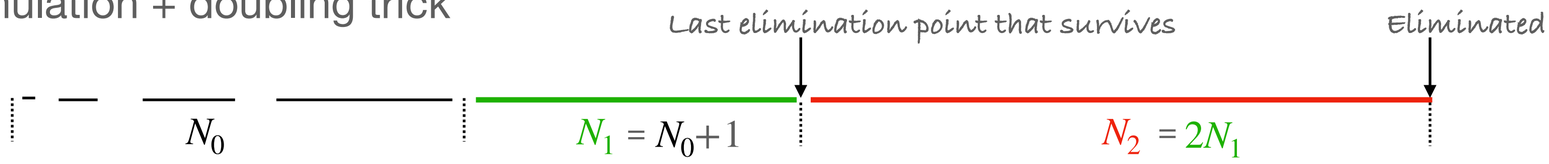
$$\mathbb{E}[R(T)] = O\left(\sum_{a \in [K]: \Delta_a > 0} \frac{\log T}{\Delta_a} + \frac{K \log T}{\varepsilon} \right)$$

Additive privacy cost

This bound is optimal

Proof intuition 💡

- **Privacy:** Laplace mech. + Parallel composition + Post-processing
- **Regret:** No noise accumulation + doubling trick



[CZ*23] Distributed Differential Privacy in Multi-Armed Bandits, ICLR'23 (Equal Contributions)

Further Applications

Tool1: Doubling & Forgetting

1. Extend to UCB?

   - Yes! Essentially the same analysis via Phased-UCB, see [LS20, ex. 7.5] [AB22]

2. Extend to other trust models as well as variants?

   - Yes! Local, distributed DP models and discrete noise, see [CZ*23]

3. Extend to linear bandits?

   - Yes! Also, local and distributed DP models, see [LZJ22]

   - **Intuition**: Phase-elimination is a perfect fit for doubling & forgetting

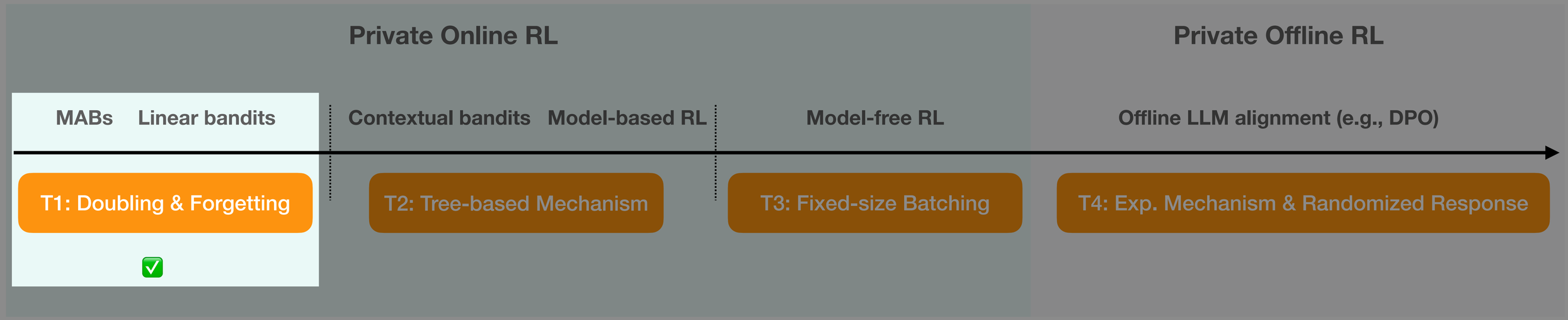
[LZJ22] Distributed Linear Bandits with Differential Privacy 🏅 Best Student Paper, WiOPT'22 & IEEE TNSE

Connections

back to private supervised learning (SL)…

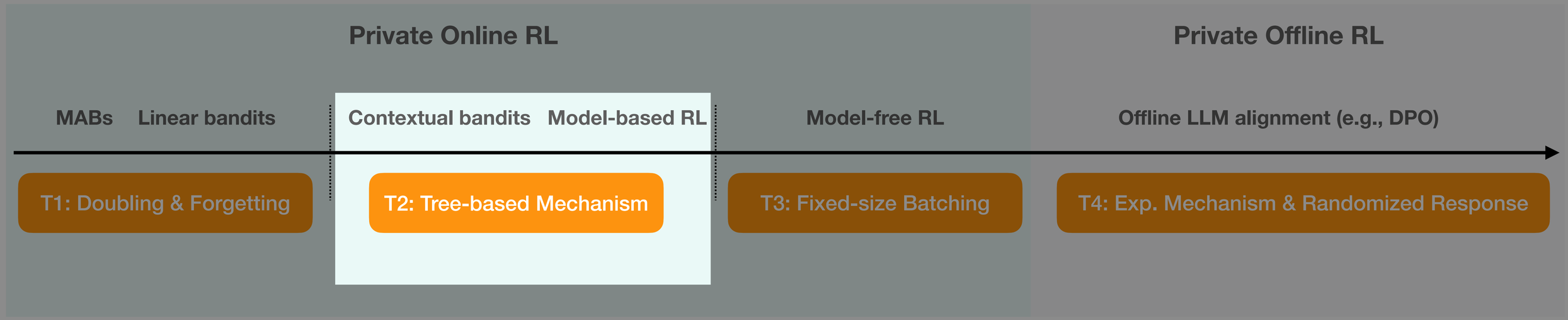
	Private SL (e.g., DP-SGD/GD)	**Private MAB** (e.g., DP-SE)	**Remark**
Privacy unit	One example/item (from one user)	One user	Both can be fixed in advance & implicitly assume “uniqueness”
Output (adversary view)	Final model weights	All T actions	DP-SGD/GD actually ensures stronger protection
Where to add noise	Gradient	Reward	Both are adaptively determined
How to bound privacy loss	Subsampling or full batch (relies on offline nature)	Parallel composition (relies on doubling trick)	The most important difference

Roadmap*



* A (biased) selection of works :-)

Roadmap*



* A (biased) selection of works :-)

Contextual Bandits

add context for personalization

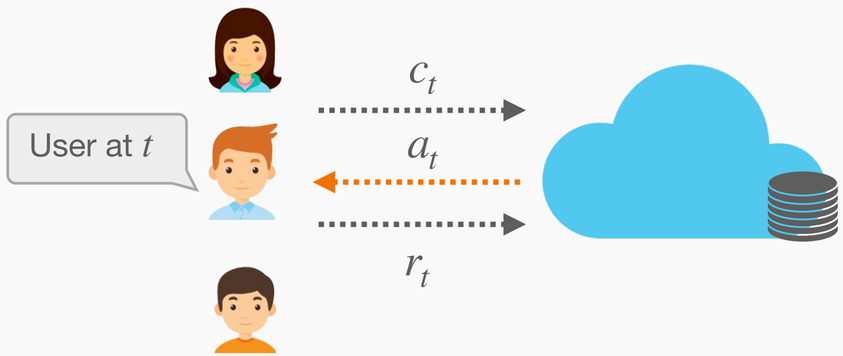
Setting

For each $t = 1, \ldots, T$:

- A user with context c_t arrives
- An action $a_t \in \mathcal{A}$ is recommended
- Reward r_t is observed, where $\mathbb{E}[r_t \mid c_t, a_t] = f^\star(c_t, a_t)$ for some unknown function $f^\star$

Goal: Minimize regret

$$R(T) := \sum_{t=1}^{T} f^\star(c^t, \pi^\star(c^t)) - \sum_{t=1}^{T} f^\star(c^t, a_t)$$



LinUCB for linear $f^\star(c, a) = \phi(c, a)^\top \theta^\star$ [APS11]

Define: $x_t := \phi(c_t, a_t)$

For $t = 1, \ldots, T$:

1. **Estimate $\theta^\star$:** $\hat{\theta}_t = V_t^{-1} U_t$,

   $(V_t = \lambda I + \sum_{s=1}^{t-1} x_s x_s^\top$ ("covariance"), $U_t = \sum_{s=1}^{t-1} x_s r_s$ ("bias") $)$

   Sufficient statistics

2. **UCB:** $a_t = \arg\max_a \phi(c_t, a)^\top \hat{\theta}_t + \beta_t \left\| \phi(c_t, a) \right\|_{V_t^{-1}}$

*There are recent general-purpose algorithms for general function classes

DP in Contextual Bandits

A challenge emerges…

Definition (DP for CB)

A contextual bandit algorithm M is (ϵ, δ)-DP if for all $D = (u_1, \ldots, u_T)$ and $D' = (u'_1, \ldots, u'_T)$, differing in one user, and for all output action sequence S,

$$\mathbb{P}[M(D) \in S] \leq e^{\varepsilon} \cdot \mathbb{P}[M(D') \in S] + \delta$$

Limitations !!

1. **Contradiction to personalization:** DP requires outputting the "same" action for two different users
   - In CB, changing one user with a different context should give a personalized action
2. **Linear regret lower bound:** DP in fact leads to a linear regret lower bound [SS18]
   - Make the problem not interesting at all
   - Need a new relaxed definition

JDP in Contextual Bandits

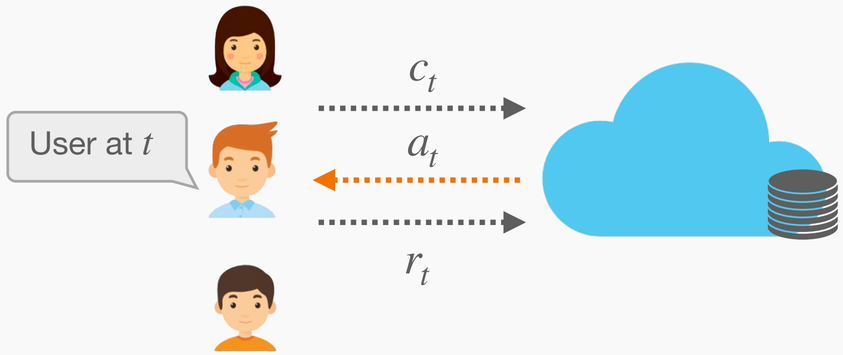
A more proper one

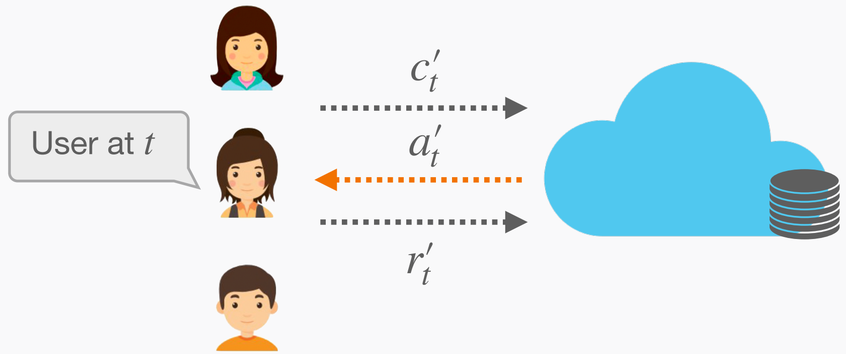
Definition (Joint DP in CB [VBKZ20])

A contextual bandit algorithm M is (ϵ, δ)-JDP if for all $D = (u_1, \ldots, u_T)$ and $D' = (u'_1, \ldots, u'_T)$, differing in one user at any t, and for all output action except round t sequence S,

$$\mathbb{P}[M_{-t}(D) \in S] \leq e^{\varepsilon} \cdot \mathbb{P}[M(D')_{-t} \in S] + \delta$$

$(a_1, \ldots, a_{t-1}, a_{t+1}, \ldots, a_T) \approx (a'_1, \ldots, a'_{t-1}, a'_{t+1}, \ldots, a'_T)$





JDP in Contextual Bandits

A more proper one

Definition (Joint DP in CB [VBKZ20])

A contextual bandit algorithm M is (ϵ, δ)-JDP if for all $D = (u_1, \ldots, u_T)$ and $D' = (u'_1, \ldots, u'_T)$, differing in one user at any t, and for all output action except round t sequence S,

$$\mathbb{P}[M_{-t}(D) \in S] \leq e^{\varepsilon} \cdot \mathbb{P}[M(D')_{-t} \in S] + \delta$$

Remarks 📝

1. **Both past and future actions:** not simply future action sequence as in [SS18]
   - This prevents colluding from future and past users
2. **Reduction to DP mechanism:** via so-called *billboard lemma* [HHR+16]
   - An Algorithm is JDP if it leverages (i) user t's own information and (ii) private signal computed via DP mechanism
   - In CB, (i) is context c_t and (ii) is all other statistics so far

Algorithm: Private-LinUCB

Tool2: Tree-based Mechanism

Non-private one

LinUCB for linear $f^\star(c,a) = \phi(c,a)^\top \theta^\star$ [APS11]

Define: $x_t := \phi(c_t, a_t)$

For $t = 1, \ldots, T$:

1. **Estimate** $\theta^\star$: $\hat{\theta}_t = V_t^{-1} U_t$,

   $(V_t = \lambda I + \sum_{s=1}^{t-1} x_s x_s^\top$ ("covariance"), $U_t = \sum_{s=1}^{t-1} x_s r_s$ ("bias"))

2. **UCB:** $a_t = \arg\max_a \phi(c_t, a)^\top \hat{\theta}_t + \beta_t \left\| \phi(c_t, a) \right\|_{V_t^{-1}}$

Sufficient statistics

All we need is to privatize these **prefix sum** statistics!

Differential Privacy 201

1. **Recall Gaussian mechanism** for private sum of l_2 bounded vectors

   i.e., $\widetilde{s}$ is the **private** sum of $\sum_{s=1}^{k} \gamma_s$ under (ϵ, δ)-DP

   $$\widetilde{s} = \sum_{s=1}^{k} \gamma_s + \mathcal{N}(0, \sigma^2 I),\ \sigma^2 \approx \frac{L^2 \log(1/\delta)}{\epsilon^2}$$

   Intuition: change one data, the sum changes in l_2, bounded by L

2. **Continual private sum** (essential for private online learning)

   i.e., stream of data $\gamma_1, \ldots, \gamma_K$, compute $\widetilde{s}_k$ for **all** k, i.e., $\sum_{s=1}^{k} \gamma_s$

   Simple Approach I: add noise ($\approx 1/\epsilon^2$) to each γ_s

   - (ϵ, δ)-DP (by post-processing)
   - total noise is K/ϵ^2 (**!**)

   Simple Approach II: add noise ($\approx 1/\epsilon^2$) to each prefix sum

   - noise is $1/\epsilon^2$ for all k
   - $\approx (\sqrt{K}\epsilon, \delta')$-DP (by advanced composition of DP)
   - i.e., for (ϵ, δ)-DP, the final total noise needs to be K/ϵ^2 (**!**)

Algorithm: Private-LinUCB

Tool2: Tree-based Mechanism

Non-private one

LinUCB for linear $f^\star(c,a) = \phi(c,a)^\top \theta^\star$ [APS11]

Define: $x_t := \phi(c_t, a_t)$

For $t = 1, \ldots, T$:

1. **Estimate** $\theta^\star$: $\hat{\theta}_t = V_t^{-1} U_t$,

   $(V_t = \lambda I + \sum_{s=1}^{t-1} x_s x_s^\top$ ("covariance"), $U_t = \sum_{s=1}^{t-1} x_s r_s$ ("bias"))

2. **UCB:** $a_t = \arg\max_a \phi(c_t, a)^\top \hat{\theta}_t + \beta_t \left\| \phi(c_t, a) \right\|_{V_t^{-1}}$

Sufficient statistics

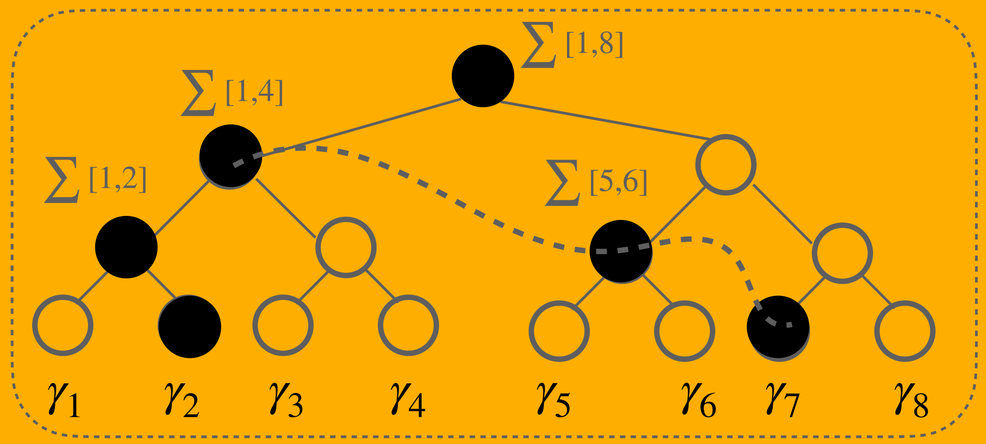
All we need is to privatize these **prefix sum** statistics!

Differential Privacy 201

Continual private sum (essential for private online learning)

i.e., a stream of data $\gamma_1, \ldots, \gamma_K$, compute $\widetilde{s}_t$ — priv. sum of $\sum_{s=1}^{k} \gamma_s$

Tree-based algorithm [CSS11]: add noise to partial sum $\sum [i,j]$



Key observations:

- each data affects at most $O(\log K)$ p-sums ($\widetilde{O}(1/\epsilon^2)$ noise each)
- each prefix sum needs at most $O(\log K)$ partial-sums (p-sums)
- total noise is still $\widetilde{O}(1/\epsilon^2)$ (✅ ignore log factor)

All the three mechanisms can be viewed as
matrix factorization mechanism, see **my blog**

Algorithm: Private-LinUCB

Tool2: Tree-based Mechanism

Non-private one

LinUCB for linear $f^\star(c,a) = \phi(c,a)^\top \theta^\star$ [APS11]

Define: $x_t := \phi(c_t, a_t)$

For $t = 1, \ldots, T$:

1. **Estimate** $\theta^\star$: $\hat{\theta}_t = V_t^{-1} U_t$,

   ($V_t = \lambda I + \sum_{s=1}^{t-1} x_s x_s^\top$ ("covariance"), $U_t = \sum_{s=1}^{t-1} x_s r_s$ ("bias"))

   Sufficient statistics

2. **UCB:** $a_t = \arg\max_a \phi(c_t, a)^\top \hat{\theta}_t + \beta_t \left\| \phi(c_t, a) \right\|_{V_t^{-1}}$

All we need is to privatize these **prefix sum** statistics!

Private one

Private-LinUCB for linear $f^\star(c,a) = \phi(c,a)^\top \theta^\star$ [SS18]

Define: $x_t := \phi(c_t, a_t)$

For $t = 1, \ldots, T$:

1. **Estimate** $\theta^\star$: $\hat{\theta}_t = \widetilde{V}_t^{-1} \widetilde{U}_t$,

   ($\widetilde{V}_t$, $\widetilde{U}_t$ are private prefix sum for V_t, U_t, obtained from Tree-based Mech.)

2. **UCB:** $a_t = \arg\max_a \phi(c_t, a)^\top \hat{\theta}_t + \beta_t \left\| \phi(c_t, a) \right\|_{\widetilde{V}_t^{-1}}$

Theoretical Guarantees

Contextual bandits

Theorem (Private-LinUCB & Lazy Version) [SS18][CZ*22]

Private-LinUCB satisfies (ϵ, δ)-JDP and achieves the following regret bound w.h.p.

$$R(T) = \widetilde{O}\left(d\sqrt{T}\right) + \widetilde{O}\left(d^{3/4}\sqrt{T\sigma}\right) \quad \text{with} \quad \sigma = \frac{\sqrt{\log(1/\delta)}}{\epsilon}$$

The same bound can be achieved with only $O(\sqrt{T})$ update via batching

Proof idea 💡

- **Privacy:** Tree-based mechanism + Billboard lemma
- **Regret:** Total noise in the prefix-sum is log order, by tree-based mechanism

[CZ*22] Shuffle Private Linear Contextual Bandits, ICML'22 (Equal Contributions)

Discussion

Tightness & other trust models

1. Is previous bound optimal (e.g., additional $\sqrt{T/\epsilon}$)?
   - the current lower bound is an additional term of d/ϵ[HZZ22]
   - under additional stochastic context condition: the best upper bound is $d^{3/2}/\epsilon$[CLRS25]
   - for general adversary context: it is still open, even without computation constraint
2. How about local DP model?
   - the first result is an additional $(dT)^{3/4}/\sqrt{\epsilon}$[ZCHLW20], but lower bound is $\sqrt{d^2T}/\epsilon$[LHG21]
   - for general (adversary) context, an exponential-time algorithm gives $\sqrt{d^3T}/\epsilon$[CR25]
   - under additional stochastic context condition: a computation-efficient algorithm gives $\sqrt{d^5T}/\epsilon$[CLRS25]
3. How about shuffle DP model?
   - the first result is $T^{3/5}/\sqrt{\epsilon}$ with only one shuffler [CZ*22]
   - it is improved to $\sqrt{T/\epsilon}$, but with $\log T$ concurrent shufflers [TKMS23]

[CZ*22] Shuffle Private Linear Contextual Bandits, ICML'22 (Equal Contributions)

Further Applications

Tool2: Tree-based Mechanism

1. How about applying it to MAB with UCB?

   - It will not yield optimal problem-dependent bound due to additional log factor

2. Extend to model-based RL?

   - Yes! For tabular MDP, see [VBKZ20], [CZ*21]

   - Yes! For linear-mixture MDP, see [Zhou22] [LGLP21]

   (The key is to find prefix-sum sufficient statistics in each setting)

3. Extend to model-free RL?

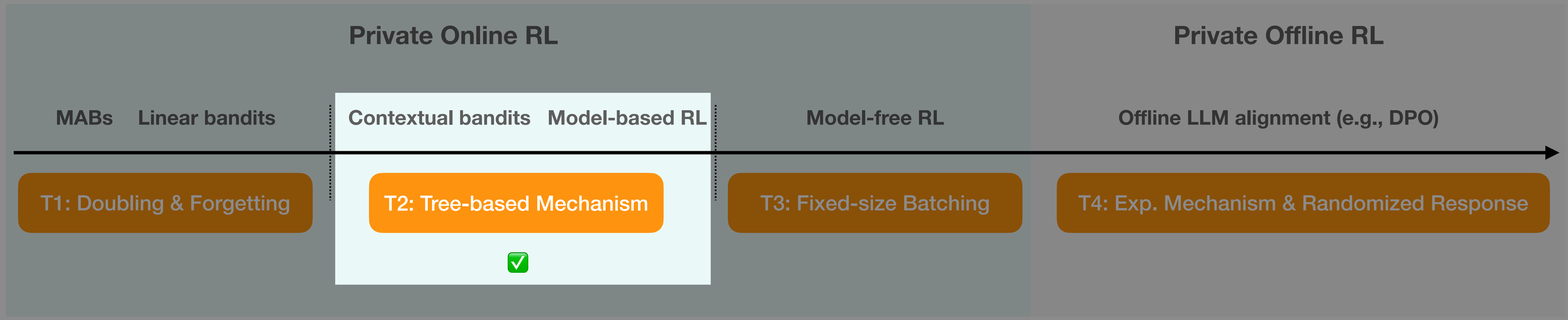
   - Unfortunately, no.

   - We no longer have the prefix-sum structure, leading to our next tool

[CZ*21] Differentially Private Regret Minimization in Episodic Markov Decision Processes, AAAI'21 (oral)

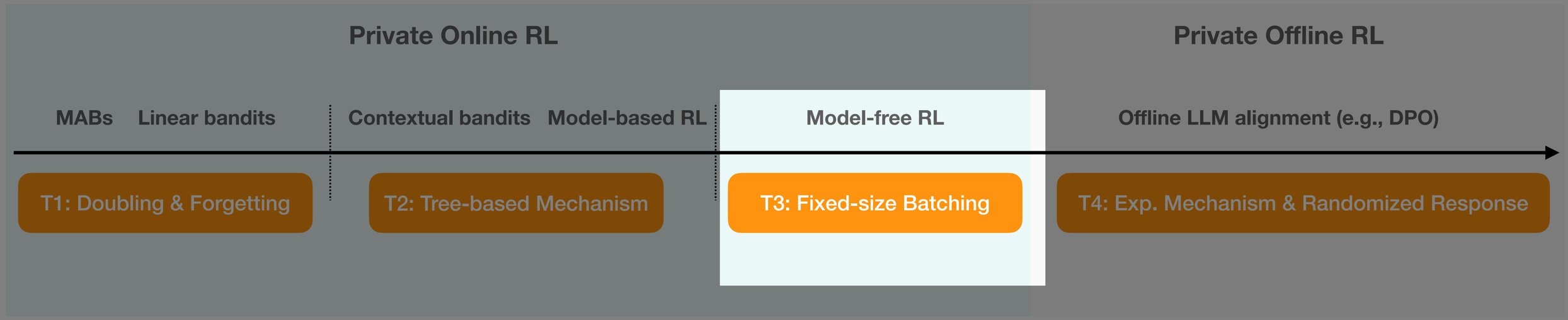
[Zhou22] Differentially Private Reinforcement Learning with Linear Function Approximation, SIGMETRICS'22

Roadmap*



* A (biased) selection of works :-)

Roadmap*



* A (biased) selection of works :-)

Model-free RL: Linear MDP

MDP & Linear MDP

An MDP is given by $M(\mathcal{S}, \mathcal{A}, H, \mathbb{P}, r)$:

- $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, H horizon length, $\mathbb{P}_h(s_{h+1} \mid s_h, a_h)$ transition prob. and $r_h(s_h, a_h)$ reward
- Value function: $V_h^\pi(s) := \mathbb{E}[\sum_{h'=h}^{H} r_{h'}(s_{h'}, \pi(s_{h'}, h')) \mid s_h = s]$ and Q-function: $Q_h^\pi(s,a) = r_h(s,a) + \mathbb{E}_{x' \sim \mathbb{P}_h(\cdot|s,a)} V_{h+1}^\pi(s')$

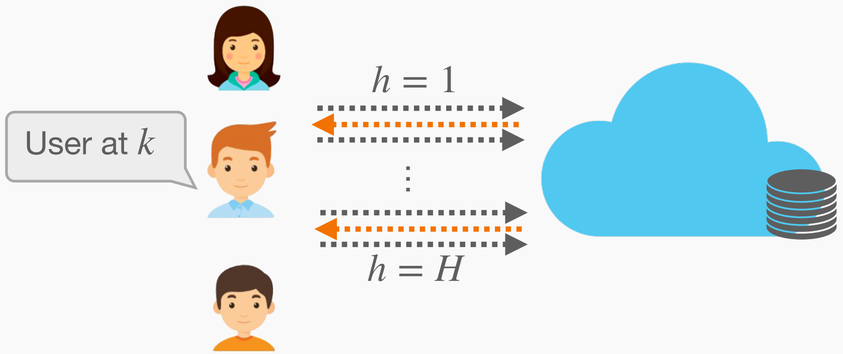
Goal: Online interact for K episodes, each with H steps to minimize the regret

$$R(K) := \sum_{k=1}^{K} V_1^\star(s_1^k) - V_1^{\pi_k}(s_1^k)$$

Linear MDP: Both transition and reward are linear mappings

Q-function will also be linear!

$$\mathbb{P}_h(\cdot \mid s,a) = \langle \phi(s,a), \mu_h(\cdot) \rangle, \quad r_h(s,a) = \langle \phi(s,a), \theta_h \rangle$$



(Can also view as sending entire trajectory in the and, given fixed policy π_k)

LSVI-UCB[JYWJ19]

Define $x_h^\tau := \phi(s_h^\tau, a_h^\tau)$

For $k \in [K], h \in [H]$:

1. **Estimation:** $w_h^k = (V_h^k)^{-1} U_h^k$

$$V_h^k = \sum_{\tau=1}^{k-1} x_h^\tau (x_h^\tau)^\top + \lambda \cdot \mathbf{I} \quad U_h^k = \sum_{\tau=1}^{k-1} x_h^\tau \left[r_h(s_h^\tau, a_h^\tau) + V_{h+1}^k(s_{h+1}^\tau) \right]$$

2. **UCB:** $Q_h^k(s,a) = \phi(s,a)^\top \hat{w}_h^k + \beta \left\| \phi(s,a) \right\|_{(V_h^k)^{-1}}, \forall s, a$

3. **Greedy:** using latest Q-function

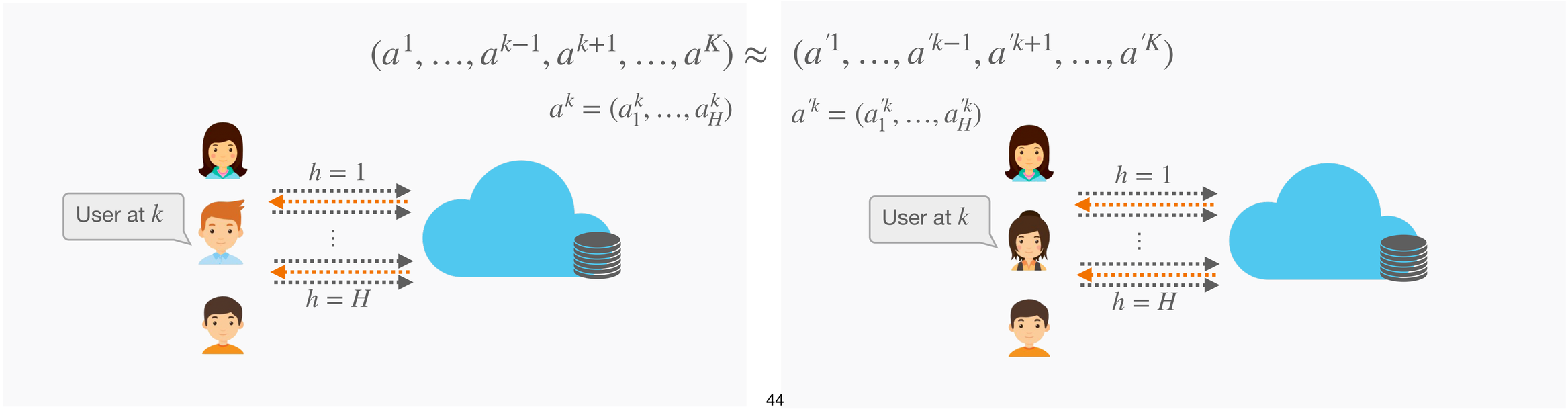
JDP in RL

lift from one-step to H step

Definition (Joint DP in RL[VBKZ20])

An RL algorithm M is (ϵ, δ)-JDP if for all $D = (u_1, \ldots, u_T)$ and $D' = (u'_1, \ldots, u'_T)$, differing in one user at any t, and for all output event except episode k, $S \subseteq \mathscr{A}^{H \times K - 1}$,

$$\mathbb{P}[M_{-k}(D) \in S] \leq e^{\varepsilon} \cdot \mathbb{P}[M(D')_{-k} \in S] + \delta$$



Algorithm: Private-LSVI-UCB

Tool3: Fixed-size Batching

Non-private one

LSVI-UCB [JYWJ19]

Define $x_h^\tau := \phi(s_h^\tau, a_h^\tau)$

For $k \in [K], h \in [H]$:

1. Estimation: $w_h^k = (V_h^k)^{-1} U_h^k$

$$V_h^k = \sum_{\tau=1}^{k-1} x_h^\tau (x_h^\tau)^\top + \lambda \cdot \mathbf{I} \quad U_h^k = \sum_{\tau=1}^{k-1} x_h^\tau \left[r_h(s_h^\tau, a_h^\tau) + V_{h+1}^k(s_{h+1}^\tau) \right]$$

prefix-sum, ✅

It is not, due to V^k !

2. UCB: $Q_h^k(s,a) = \phi(s,a)^\top \hat{w}_h^k + \beta \left\| \phi(s,a) \right\|_{(V_h^k)^{-1}}, \forall s, a$

3. Greedy: using latest Q-function

Private one

Private-LSVI-UCB [LGLP21]

For $k \in [K], h \in [H]$:

Batching update with noise

If $k \,\%\, B = 0$ do update:

1. Estimation: $w_h^k = (\widetilde{V}_h^k)^{-1} \widetilde{U}_h^k$

$\widetilde{V}_h^k$ is obtained via tree-based mechanism

$$\widetilde{U}_h^k = \sum_{\tau=1}^{k-1} x_h^\tau \left[r_h(s_h^\tau, a_h^\tau) + V_{h+1}^k(s_{h+1}^\tau) \right] + \mathcal{N}(0, \sigma^2 I)$$

2. UCB: $Q_h^k(s,a) = \phi(s,a)^\top \hat{w}_h^k + \beta \left\| \phi(s,a) \right\|_{(V_h^k)^{-1}}, \forall s, a$

3. Greedy: using latest Q-function

Theoretical Guarantees

Linear MDP

Theorem (Private-LSVI-UCB) [LGLP21]

Private-LSVI-UCB with $\sigma^2 \approx_\delta \frac{K}{\epsilon^2 B}$ satisfies (ϵ, δ)-JDP and attains regret w.h.p.

$$R(T) \lesssim_\delta \text{poly}(H, d)\left(\sqrt{K} + K^{3/5}/\epsilon^{2/5}\right)$$

Proof idea 💡

- **Privacy:** Tree-based mechanism + Gaussian mechanism + Advanced composition

  - The "dominated" term is $\widetilde{U}^k$, giving the noise σ^2 above via advanced composition over T/B updates

- **Regret:** A generic regret bound under batching with noise, see [CZ*22]

  $$R(T) \lesssim \text{poly}(H, d)\left(B + d\sqrt{K} + \sqrt{\sigma_0 K}\right)$$ where σ_0 is the total noise in the sufficient statistics

  In our case, $\sigma_0^2 = \sigma^2$ above, giving regret with optimal choice of B

[CZ*22] Shuffle Private Linear Contextual Bandits, ICML'22 (Equal Contributions)

Can we do better?

Adaptive lazy update fails…

- In the non-private, one can use determinant-trick for adaptive lazy update
  - the total number of update is now log order
  - which seems to reduce the privacy cost due to composition
- However, in the private case, tricky things happen
  - the adaptive condition also needs privacy protection
  - which invalidates standard determinant trick
  - this leads to proof gaps in several existing works

Further Applications

Tool3: Fixed-size Batching

1. Useful for shuffle model

   - leveraging it, we give the first bound under shuffle DP, see [CZ*22]

   - essentially based on the previous generic regret bound under batching with noise

2. Useful for federated contextual bandits (CB)

   - leveraging it, we give the first correct regret bound for private federated CB

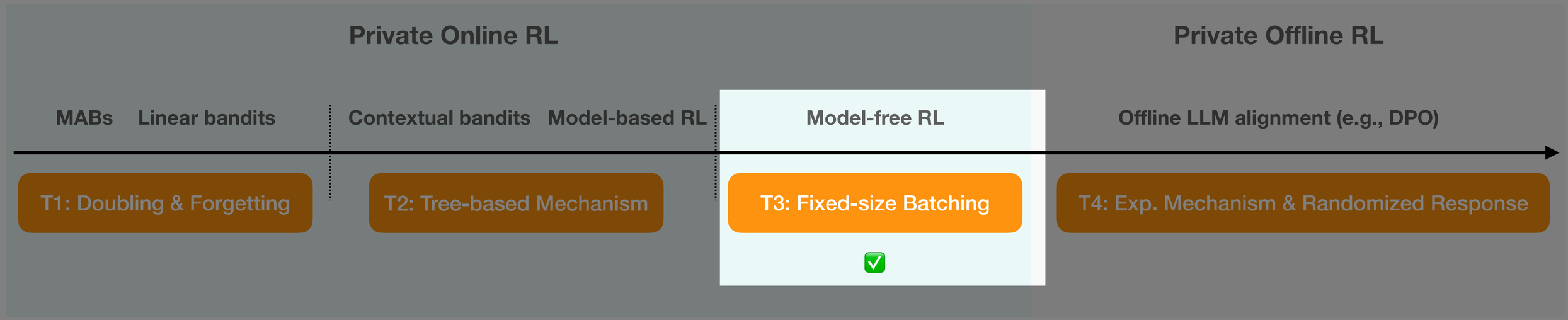
   - again, the issues are due to adaptive lazy update

3. Useful for RL with general function approximations

   - our ongoing work:-)

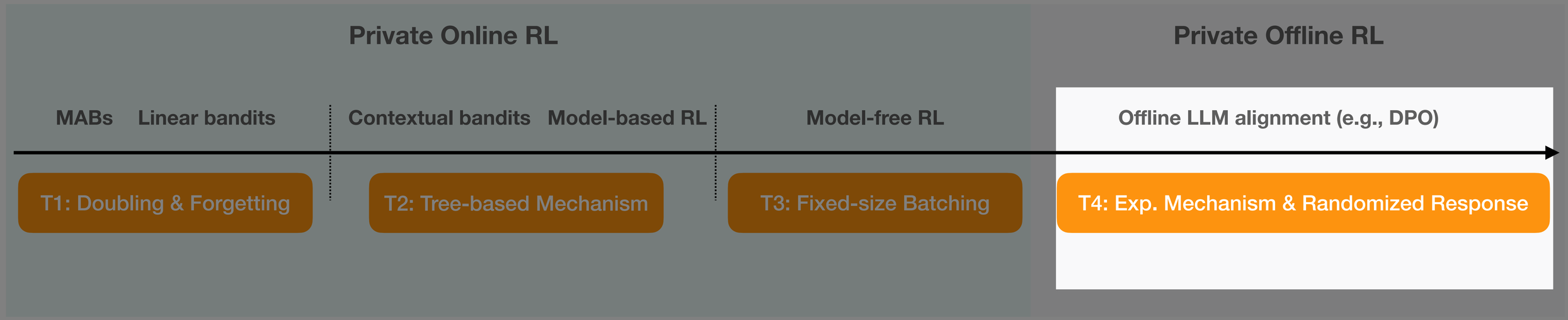
[CZ*22] Shuffle Private Linear Contextual Bandits, ICML'22 (Equal Contributions)

Roadmap*



* A (biased) selection of works :-)

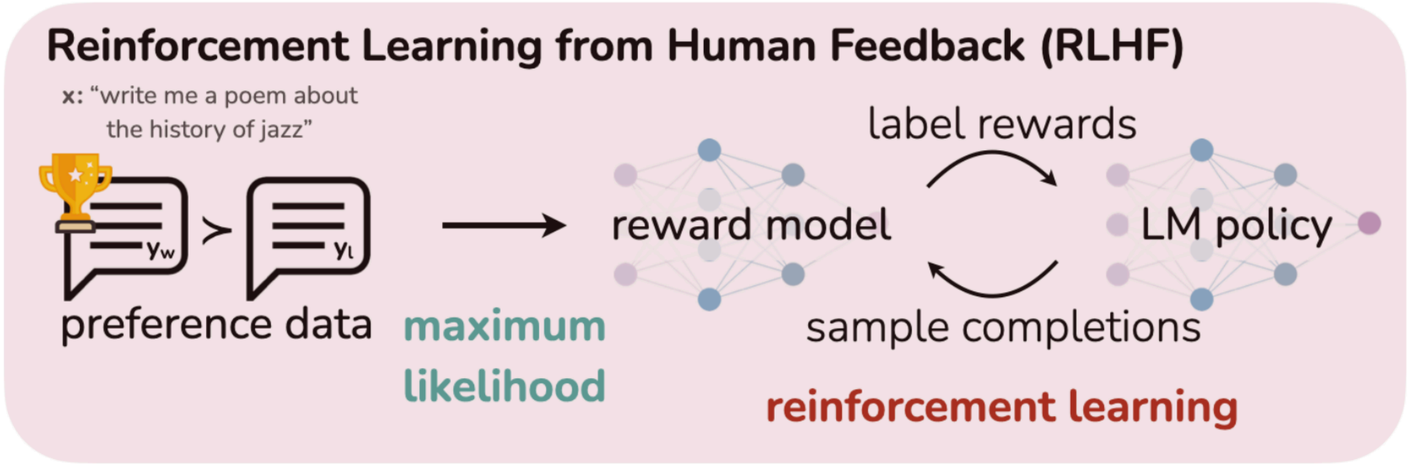
Roadmap*

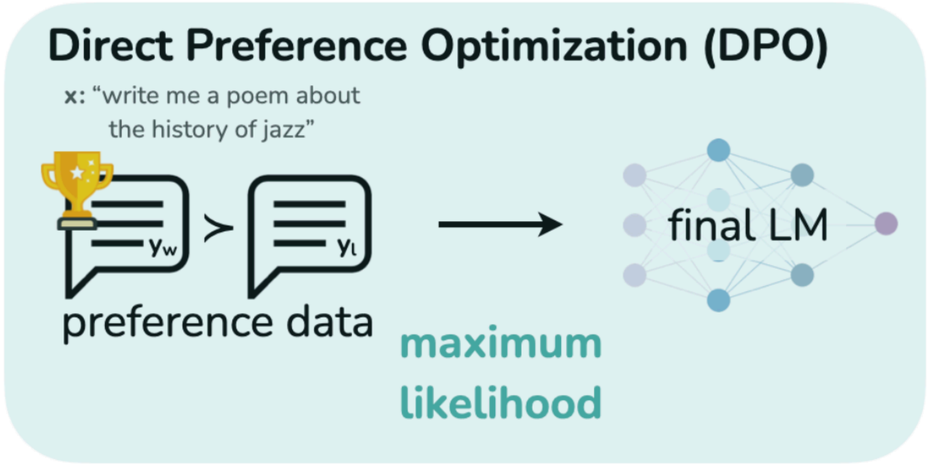


* A (biased) selection of works :-)

LLM Alignment

Align outputs with human values





Two Popular Paradigms in LLM Alignment (figure from Rafailov et al. 2023)

LLM Alignment

Align outputs with human values

Formal Setting

Given an **offline** preference dataset $D_{\text{pref}} = \{x_i, a_i^0, a_i^1, y_i\}_{i=1}^n$

- $x_i \sim \rho$, i.i.d and $a_i^0 \sim \pi_{\text{ref}}(\cdot \mid x_i)$, $a_i^1 \sim \pi_{\text{ref}}(\cdot \mid x_i)$
- $y_i \in \{0,1\} \sim \text{Ber}(P(a_i^1 \succ a_i^0 \mid x_i))$
- $P(a_i^1 \succ a_i^0 \mid x_i)$ satisfies BT-preference model: $P(a_i^1 \succ a_i^0 \mid x_i) = \dfrac{\exp(r^\star(x_i, a_i^1))}{\exp(r^\star(x_i, a_i^1)) + \exp(r^\star(x_i, a_i^0))}$

Goal: Minimize sub-optimality gap: $\text{SubOpt}(\hat{\pi}, \pi^\star) := J(\pi^\star) - J(\hat{\pi})$, with $J(\pi) := \mathbb{E}_{x\sim\rho, y\sim\pi(\cdot|x)}[r^\star(x, y)]$

Direct Preference Optimization (DPO)

Solve: $\hat{\pi}_{\text{DPO}} = \arg\max_{\pi\in\Pi} \sum_{(x,a_+,a_-)\in D_{\text{pref}}} \log\left[\sigma\left(\beta h_{\text{DPO}}(x, a_+, a_-)\right)\right]$

- $h_{\text{DPO}}(x, a_+, a_-) := \log \dfrac{\pi(a_+ \mid x)}{\pi_{\text{ref}}(a_+ \mid x)} - \log \dfrac{\pi(a_- \mid x)}{\pi_{\text{ref}}(a_- \mid x)}$
- $\sigma(z)$ is sigmoid
- $a_+ = a^y$ (preferred one)
- β is some regularization parameter

χ^2-Preference Optimization (χPO) [HZXLSKF25]

Solve: $\hat{\pi}_{\chi\text{PO}} = \arg\max_{\pi\in\Pi} \sum_{(x,a_+,a_-)\in D_{\text{pref}}} \log\left[\sigma\left(\beta h_{\chi\text{PO}}(x, a_+, a_-)\right)\right]$

- $h_{\chi\text{PO}}(x, a_+, a_-) := \phi\left(\dfrac{\pi(a_+ \mid x)}{\pi_{\text{ref}}(a_+ \mid x)}\right) - \phi\left(\dfrac{\pi(a_- \mid x)}{\pi_{\text{ref}}(a_- \mid x)}\right)$
- $\phi(u) := u + \log u$

This additional term introduces **pessimism** — key for single-policy concentrability

Privacy in LLM Alignment

Local & Central Models

Definition (Randomized Response & LDP)

The true preference label y is passed through local randomized response $\mathscr{R}$, generating $\tilde{y}$ with

$$\mathbb{P}[\tilde{y} = y] = \frac{e^{\varepsilon}}{1 + e^{\varepsilon}} \quad \text{and} \quad \mathbb{P}[\tilde{y} \neq y] = \frac{1}{1 + e^{\varepsilon}}$$

This satisfies local ϵ-label-DP

Essentially standard LDP definition applies to label

Definition (Central DP)

An offline alignment $\mathscr{A}$ is (ϵ, δ)-DP if for all S,

$$\mathbb{P}[\mathscr{A}(D_{\text{pref}}) \in S] \leq e^{\varepsilon} \cdot \mathbb{P}[\mathscr{A}(D'_{\text{pref}}) \in S] + \delta$$

holds for any pair $(D_{\text{pref}}, D'_{\text{pref}})$, differing in one sample (x_i, a_i^0, a_i^1, y_i)

Essentially standard DP definition

Algorithm: SquareχPO

from log-loss to square loss

Non-private one

χ^2-Preference Optimization (χPO)

Solve: $\hat{\pi}_{\chi\text{PO}} = \arg\max_{\pi \in \Pi} \sum_{(x,a_+,a_-) \in D_{\text{pref}}} \log \left[\sigma \left(\beta h_{\chi\text{PO}}(x, a_+, a_-) \right) \right]$

- $h_{\chi\text{PO}}(x, a_+, a_-) := \phi \left(\frac{\pi(a_+ \mid x)}{\pi_{\text{ref}}(a_+ \mid x)} \right) - \phi \left(\frac{\pi(a_- \mid x)}{\pi_{\text{ref}}(a_- \mid x)} \right)$
- $\phi(u) := u + \log u$

Private one in local model

View it as a debiased loss

SquareχPO

Solve: $\hat{\pi} \leftarrow \arg\min_{\pi \in \Pi} \sum_{i \in [n]} \left[2\sigma \left(\beta h_{\chi\text{PO},i} \right) - 1 - c(\varepsilon) z_i \right]^2$

- $h_{\chi\text{PO},i} := \phi \left(\frac{\pi(a_i^1 \mid x_i)}{\pi_{\text{ref}}(a_i^1 \mid x_i)} \right) - \phi \left(\frac{\pi(a_i^0 \mid x_i)}{\pi_{\text{ref}}(a_i^0 \mid x_i)} \right)$
- $c(\epsilon) := \frac{e^\epsilon + 1}{e^\epsilon - 1}$ and $z_i = 2\widetilde{y}_i - 1$

Algorithm: SquareχPO

from log-loss to square loss

Private one in local model

SquareχPO

Solve: $\hat{\pi} \leftarrow \arg\min_{\pi \in \Pi} \sum_{i \in [n]} \left[2\sigma\left(\beta h_{\chi\text{PO},i}\right) - 1 - c(\varepsilon) z_i \right]^2$

- $h_{\chi\text{PO},i} := \phi\left(\frac{\pi(a_i^1 \mid x_i)}{\pi_{\text{ref}}(a_i^1 \mid x_i)}\right) - \phi\left(\frac{\pi(a_i^0 \mid x_i)}{\pi_{\text{ref}}(a_i^0 \mid x_i)}\right)$
- $c(\epsilon) := \frac{e^\epsilon + 1}{e^\epsilon - 1}$ and $z_i = 2\widetilde{y}_i - 1$

Private one in central model

SquareχPO

Sample $\hat{\pi}$ from Π using exponential mechanism with probability

$$P(\pi) \propto \exp\left(-\frac{\varepsilon}{8} \cdot L(\pi; D_{\text{pref}})\right)$$

- $L(\pi; D_{\text{pref}}) := \sum_{i \in [n]} \left[2\sigma\left(\beta h_{\chi\text{PO},i}\right) - 1 - z_i \right]^2$

Theoretical Guarantees

offline alignment

Theorem (SquareχPO)[ZWWO25]

Under local model, whp $1-\beta$, SquareχPO attains

$$\mathrm{SubOpt}(\hat{\pi}, \pi^\star) \lesssim \kappa(\pi^\star) \cdot \left(c(\varepsilon) \sqrt{\frac{\log(|\Pi|/\beta)}{n}} \right), \text{ where } c(\epsilon) := \frac{e^\epsilon + 1}{e^\epsilon - 1}$$

Optimal scaling

Under central model, whp $1-\beta$, SquareχPO attains

$$\mathrm{SubOpt}(\hat{\pi}, \pi^\star) \lesssim \kappa(\pi^\star) \cdot \left(\left(1 + 1/\sqrt{\epsilon}\right) \sqrt{\frac{\log(|\Pi|/\beta)}{n}} \right)$$

Single-policy concentrability, i.e., only depends on the comparator policy

Proof idea 💡

- **Local model:** Debiased estimator under randomized response + generalization of LS under local privacy
- **Central model:** Exponential mechanism + generalization of LS under central privacy

[ZWWO] SquareχPO: Differentially Private and Robust χ2-Preference Optimization in Offline Direct Alignment, ICML'25

Further Applications

Tool4: Exp. Mechanism & Randomized Response

1. What if there exists label corruption besides privacy?

   - Yes! SquareχPO will still work under additional Huber corruption

   - For local privacy, the order of corruption and privacy protection leads to separation result

2. What if we get rid of BT-preference model?

   - Yes! A variant of SquareχPO will still work under general preference model

3. What if it is strong adaptive corruption?

   - Yes! The same debiased loss can handle interplay between privacy and corruption [ZWO25]

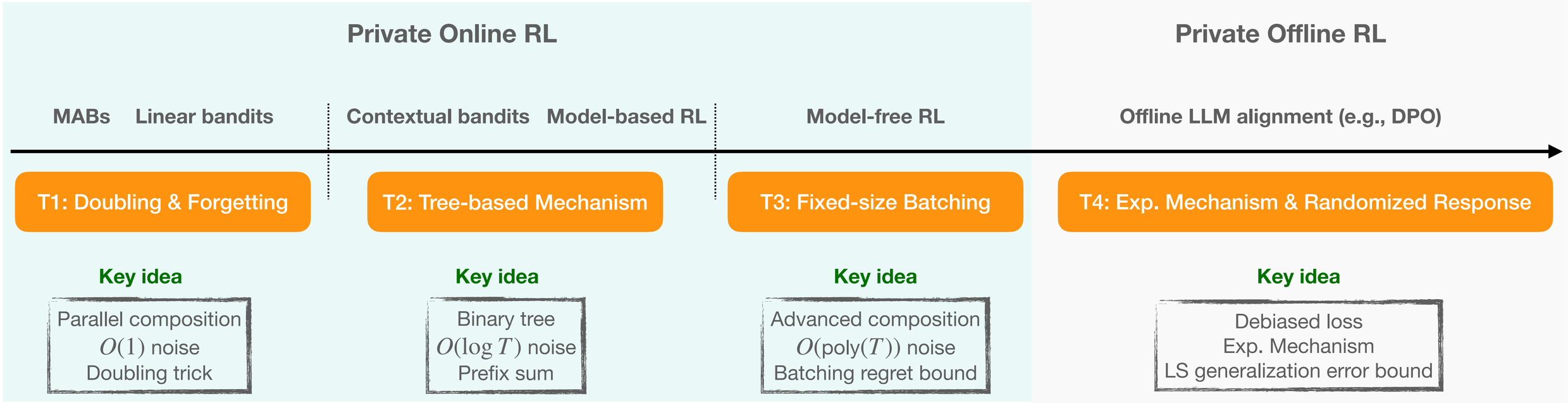
   - But, it is currently only for linear function approximations

4. What if we choose RLHF rather than DPO?

   - Yes! a unified analysis of RLHF and DPO under linear function approximation [ZWO25]

[ZWO25] A Unified Theoretical Analysis of Private and Robust Offline Alignment: from RLHF to DPO, ICML'25 (Spotlight)

Summary



* A (biased) selection of works :-)

Open Problems

Open & Important Problems

a biased selection

1. How about private RL with general function approximations?
    - The non-private case has advanced quickly recently
    - Given the lack of prefix-sum, the first approach is fixed-size batching
    - Can we reduce to private supervised and online learning?
2. What's the complexity measure for private RL learnability?
    - For non-private RL, there are several recent measures, e.g., DEC, GEC, SEC
    - For private PAC learning, we know it is **Littlestone dimension** (hence online learning)
    - Very recently, [CR25] shows that one measure is **fractional covering number**, but exponential gap remains
3. How about private RL for LLM (e.g., alignment and reasoning)?
    - how to define privacy?
    - Outcome reward vs. per-step reward
4. What's the interplay of privacy with robustness and fairness in private RL?

My Wonderful Collaborators*

Thank you!

Sayak Ray Chowdhury, Indian Institute of Technology Kanpur
Wenbo Ren, OSU
Jia Liu, OSU
Ness Shroff, OSU
Bo Ji, Virginia Tech
Duo Cheng, Virginia Tech
Francesco Orabona, KAUST
Di Wang, KAUST
Yulian Wu, KAUST
Changyu Gao, UW-Madison
Andrew Lowy, UW-Madison
Stephen Wright, UW-Madison
Nagarajan Natarajan, Microsoft Research, India
Jian Tan, Alibaba USA
Wei Zhang, TAMU

* Only include co-authors on papers related with privacy

Thank you!