42nd International Conference on Machine Learning (ICML2025)

Algorithmic Recourse for Long-Term Improvement

Kentaro Kanamori[1], Ken Kobayashi[2], Satoshi Hara[3], Takuya Takagi[1]

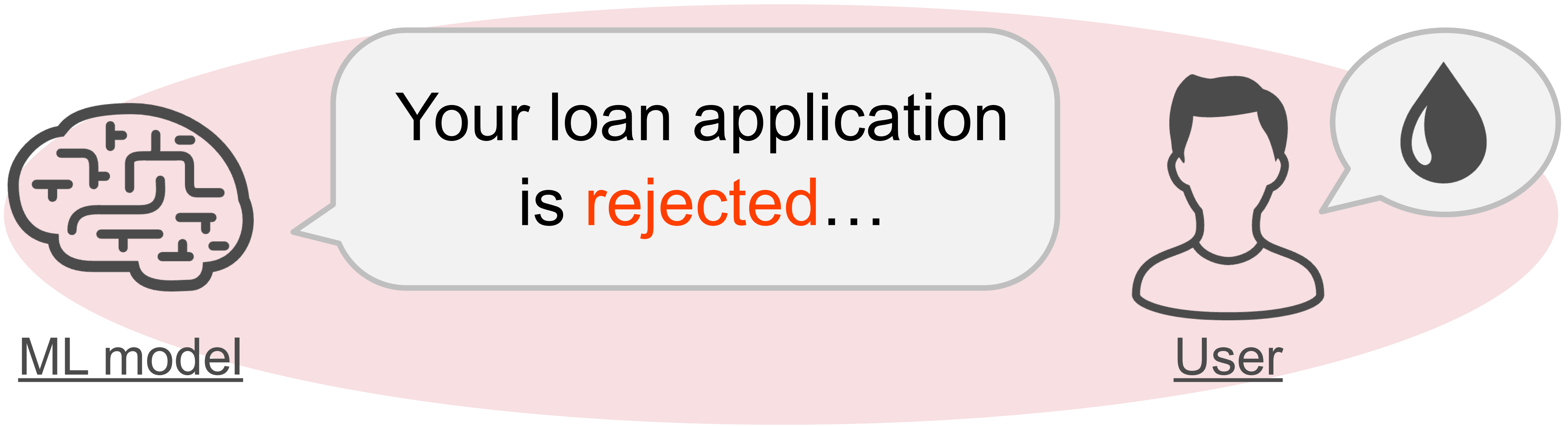
[1]Fujitsu Limited, [2]Institute of Science Tokyo, [3]The University of Electro-Communications

Background

Algorithmic recourse aims to provide an "action" for altering unfavorable predictions

Algorithmic Recourse [Ustun+ 19]

Explaining a "*recourse action*" for obtaining a favorable prediction result from an ML model

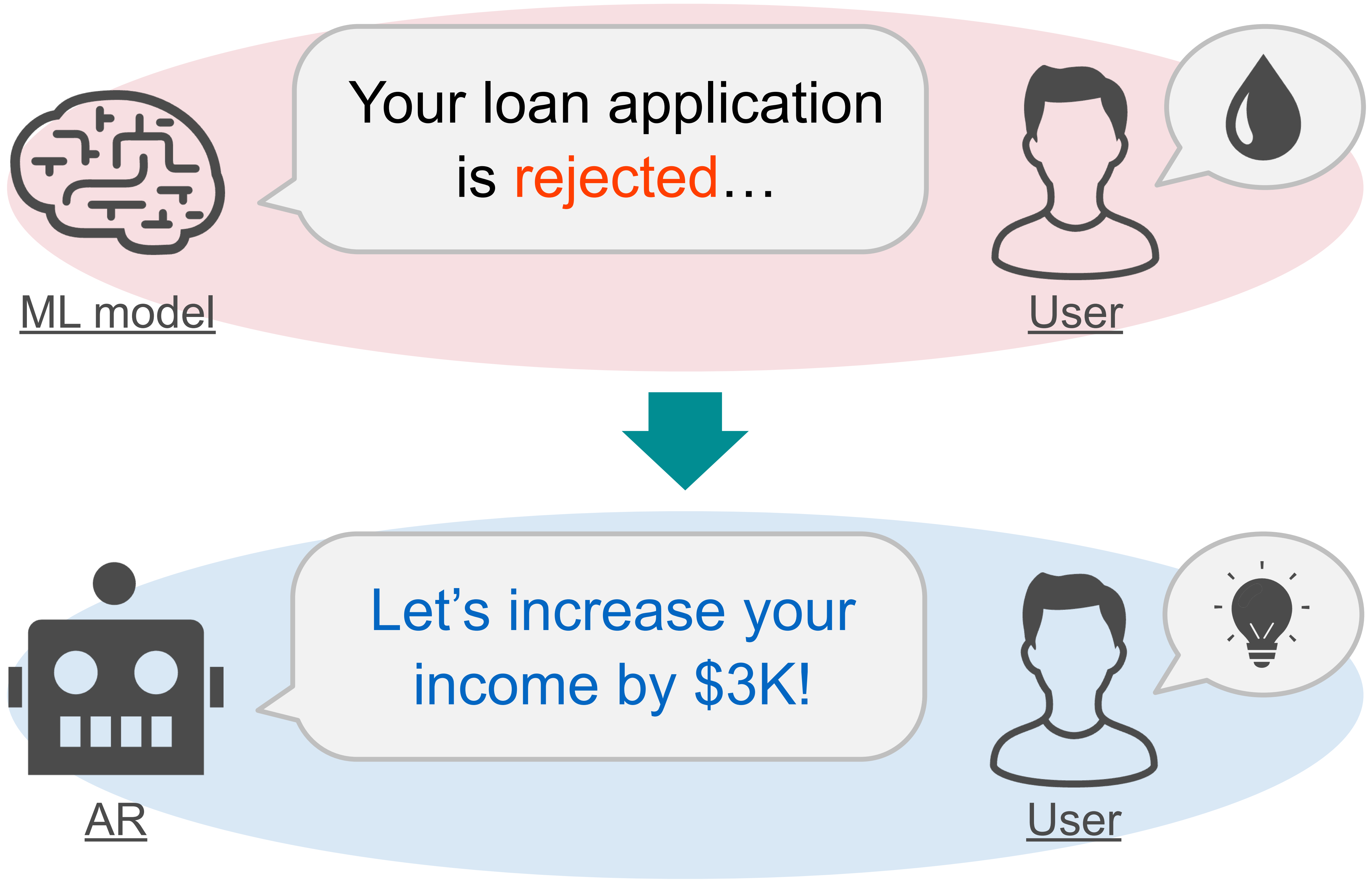


Background

Algorithmic recourse aims to provide an "action" for altering unfavorable predictions

Algorithmic Recourse [Ustun+ 19]

Explaining a "*recourse action*" for obtaining a favorable prediction result from an ML model



Problem 1. (Algorithmic Recourse; AR)

Given a model $h: \mathscr{X} \to \mathscr{Y}$, an input instance $x \in \mathscr{X}$, and a favorable class $y^* \in \mathscr{Y}$, find an action a^* such that

$$a^* = \arg\min_{a \in \mathscr{A}} c(a \mid x) \quad \text{s.t.} \quad h(x+a) = y^*$$

where $\mathscr{A}$ is a set of feasible actions and c is a cost function.

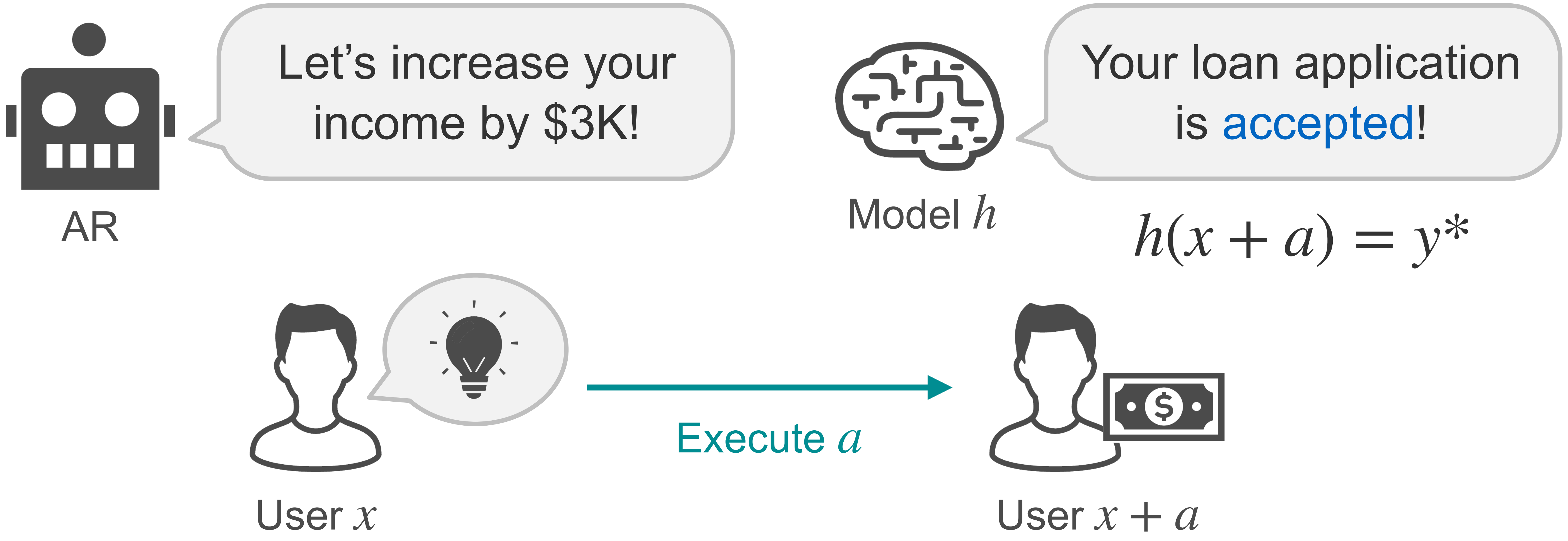
- Provide a minimum-cost action a that alters the prediction by the ML model h

Motivation

Provide improvement-oriented actions for making the real-world outcome better

"Improvement" [König+ 23]

To maintain the quality and reliability of high-stakes decision-making tasks,
we need to provide actions that improve the user's real-world outcome as well as prediction

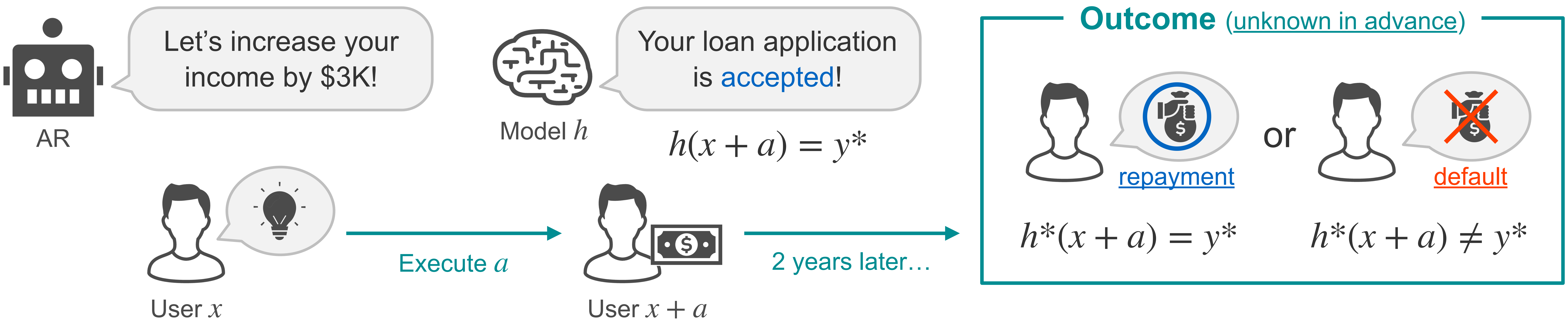


Motivation

Provide improvement-oriented actions for making the real-world outcome better

“Improvement” [König+ 23]

To maintain the quality and reliability of high-stakes decision-making tasks,
we need to provide actions that <u>improve the user’s real-world outcome</u> as well as prediction

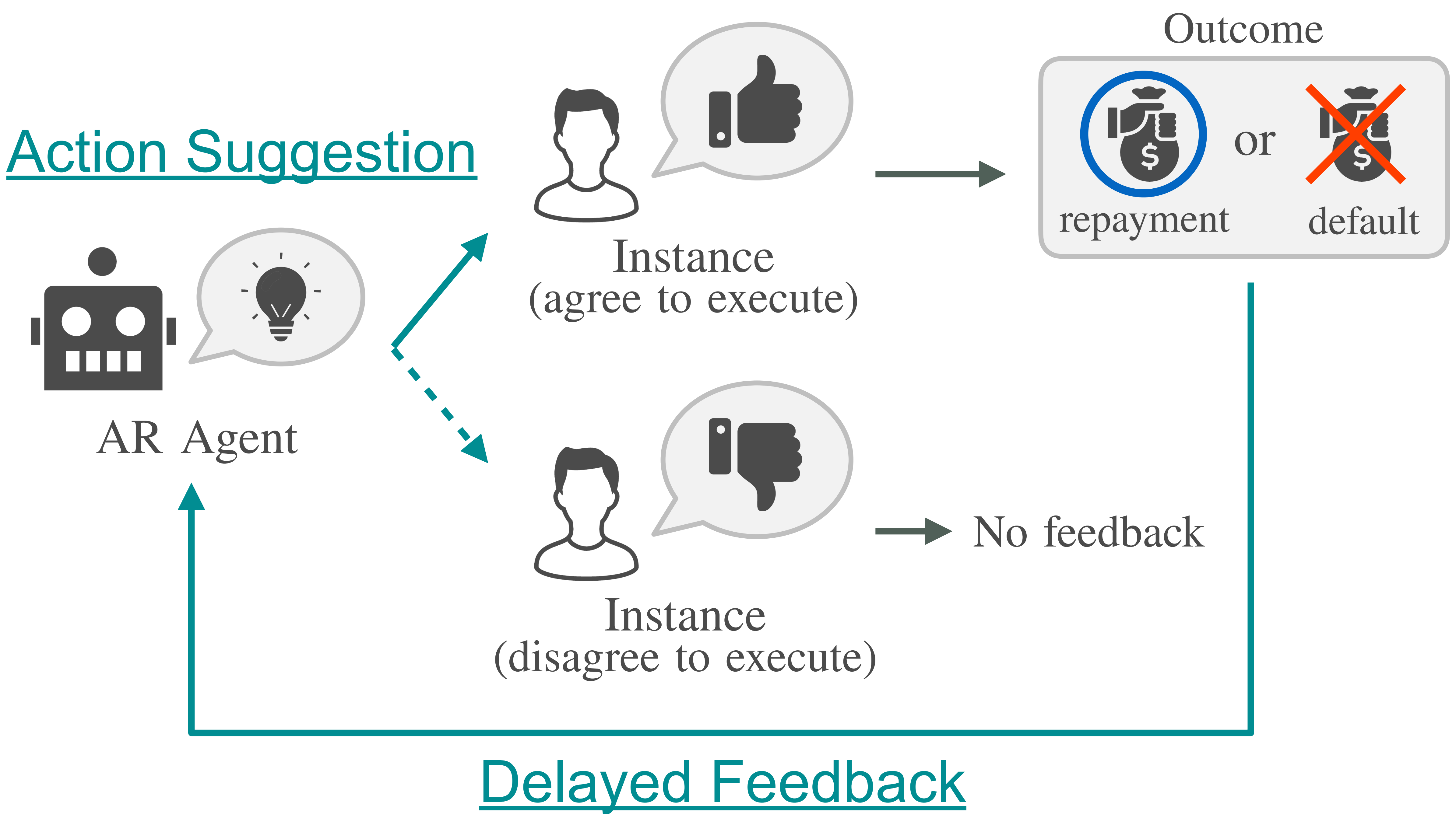


- Achieving improvement is fundamentally difficult because we do not know the oracle h^*

Problem Formulation

Suggest actions for given instances and observe delayed feedback on outcomes

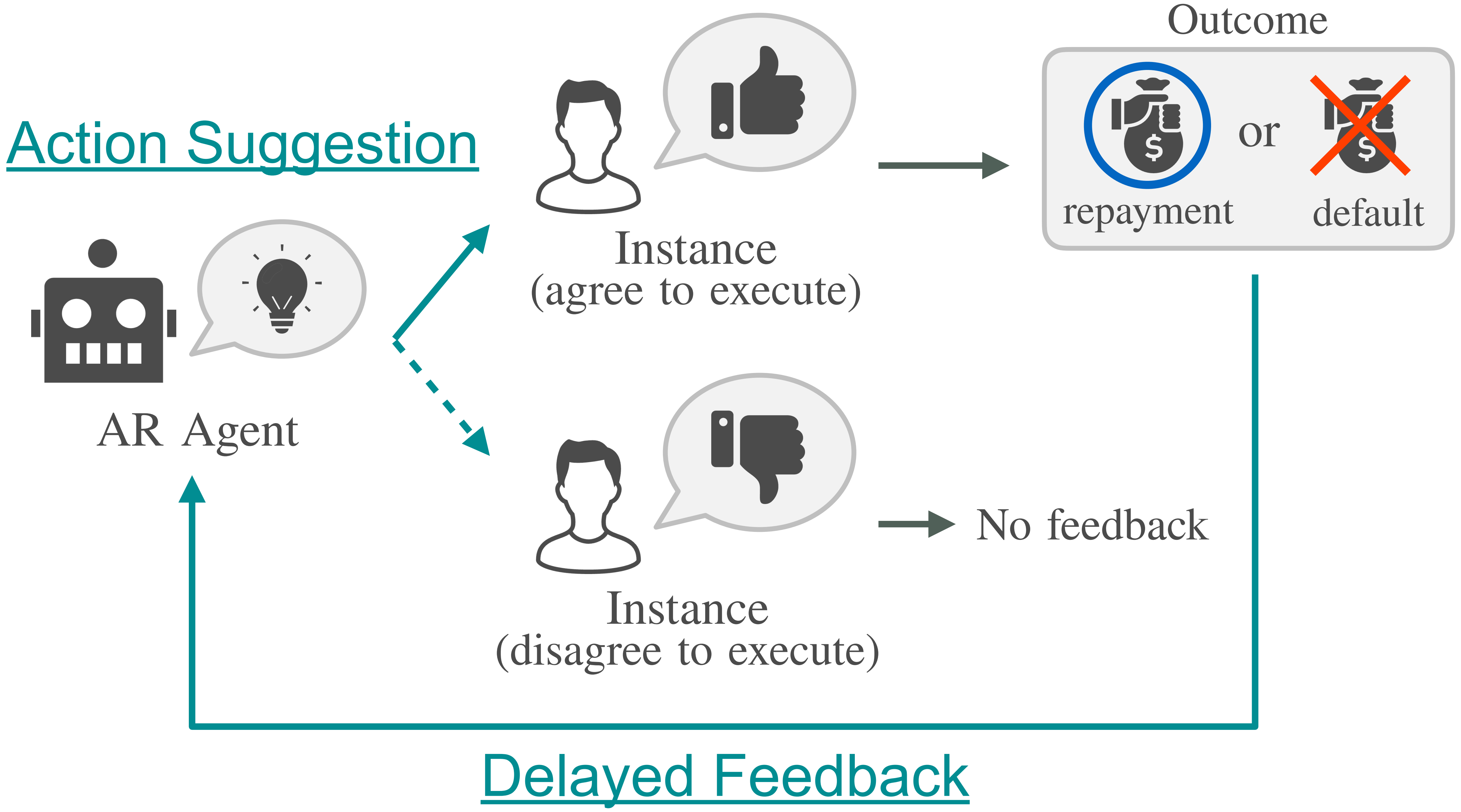
Assumption We can observe the outcome if an instance x executes a suggested action a



Problem Formulation

Suggest actions for given instances and observe delayed feedback on outcomes

Assumption We can observe the outcome if an instance x executes a suggested action a



Problem 2. (AR for Long-Term Improvement; ARLIM)

For each round $t = 1,2,\ldots,T,$

1. Receive an instance x_t and candidate valid actions $\mathcal{A}_t$
2. Suggest an action $a_t \in \mathcal{A}_t$ based on the past observations
3. Sample a reward $R_t \sim \mathcal{B}(p_t)$ and delay $D_t \sim \mathcal{D}$, where p_t is the probability that x_t executes a_t and $h^*(x_t + a_t) = y^*$
4. Observe feedback on the past rewards $\{R_s \mid s + D_s = t\}_{s=1}^{t-1}$

Goal Maximize the mean expected reward $R_T = \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[R_t]$

▸ We aim to provide improvement-oriented actions a_t for as many instances x_t as possible

Algorithms

Apply contextual linear bandit and contextual Bayesian optimization algorithms

Contextual Linear Bandit (CLB)

Assumption We can model the probability p_t as:

$$p_t = \underbrace{\exp(-c(a_t \mid x_t))}_{\text{prob. of execution (known)}} \cdot \underbrace{\mathbb{P}(h^*(x_t + a_t) = y^*)}_{\text{prob. of improvement (unknown)}}$$

- Our problem can be reduced to the *CLB problem under stochastic delayed feedback* [Vernade+ 20]

Proposition 4.2

There exists an algorithm (*LinUCB*) that satisfies:

$$R_T \geq \underbrace{\frac{1}{T}\sum_{t=1}^{T} R_t^*}_{\text{optimal rewards}} - \underbrace{\mathcal{O}\left(\log T/\sqrt{T}\right)}_{\text{converges to 0}}$$

Algorithms

Apply contextual linear bandit and contextual Bayesian optimization algorithms

Contextual Linear Bandit (CLB)

Assumption We can model the probability p_t as:

$$p_t = \underbrace{\exp(-c(a_t \mid x_t))}_{\text{prob. of execution (known)}} \cdot \underbrace{\mathbb{P}(h^*(x_t + a_t) = y^*)}_{\text{prob. of improvement (unknown)}}$$

prob. of execution (**known**)

prob. of improvement (**unknown**)

- Our problem can be reduced to the *CLB problem under stochastic delayed feedback* [Vernade+ 20]

Proposition 4.2

There exists an algorithm (*LinUCB*) that satisfies:

$$R_T \geq \underbrace{\frac{1}{T}\sum_{t=1}^{T} R_t^*}_{\text{optimal rewards}} - \underbrace{\mathcal{O}\left(\log T/\sqrt{T}\right)}_{\text{converges to 0}}$$

optimal rewards

converges to 0

Contextual Bayesian Optimization (CBO)

Idea Train a model $f\colon \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ such that

$$f(x_t, a_t) \approx R_t$$

using the past observations $Z_t = \{(x_s, a_s, R_s)\}_{s=1}^{t-1}$

- Our problem can be regarded as the *CBO problem under stochastic delayed feedback* [Verma+ 22]

😊 No need for the cost function c to be known

😥 Scalability issue with the GP-based algorithms

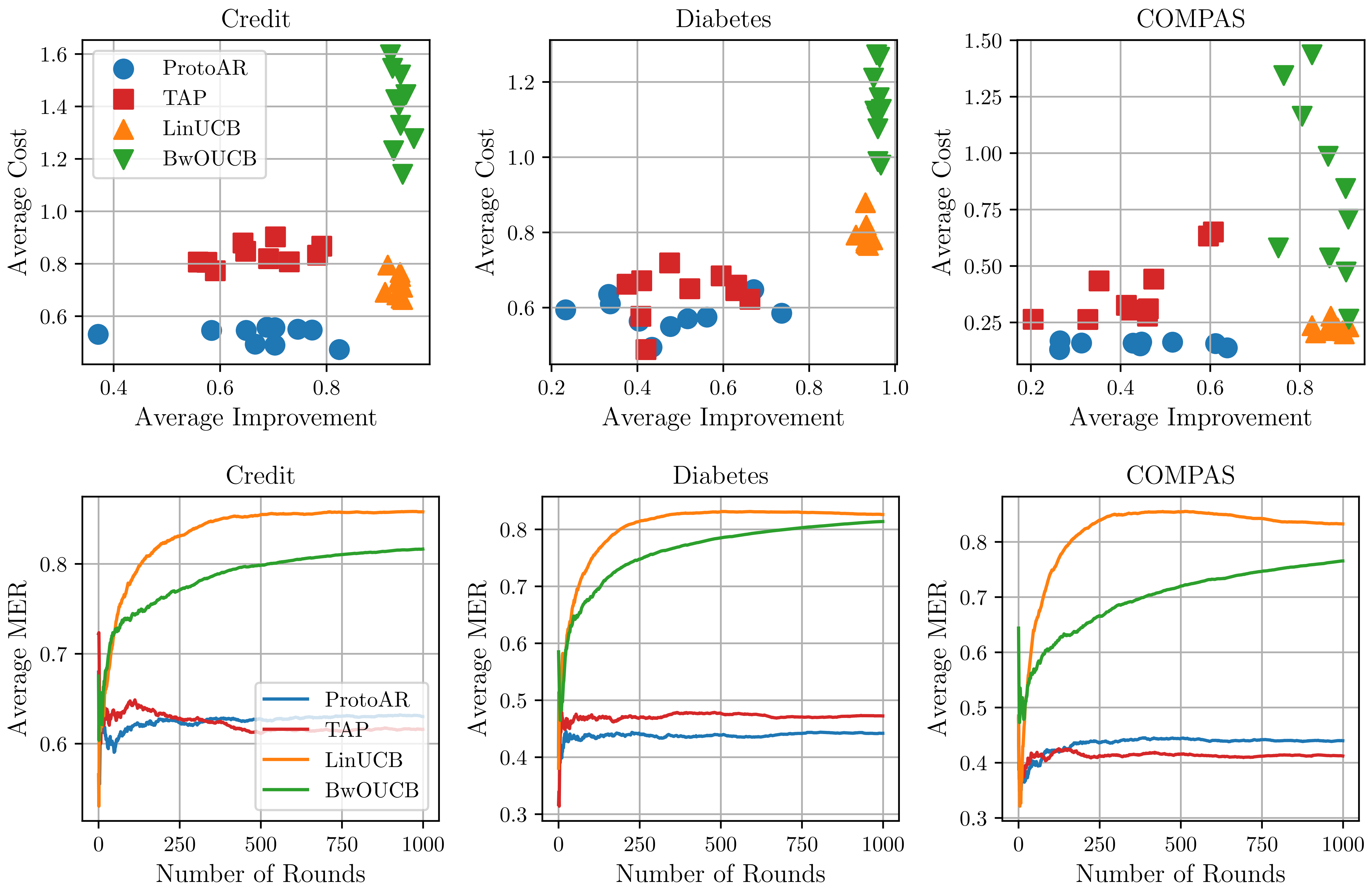
- By employing *BwO forest* [Kim+ 22] instead of GP, we propose a scalable algorithm (*BwOUCB*)

Experiments

Achieve higher improvement than baselines without significantly degrading cost

Case 1. "Noiseless" Cost Scenario

- Our LinUCB attained higher improvements while maintaining comparable costs

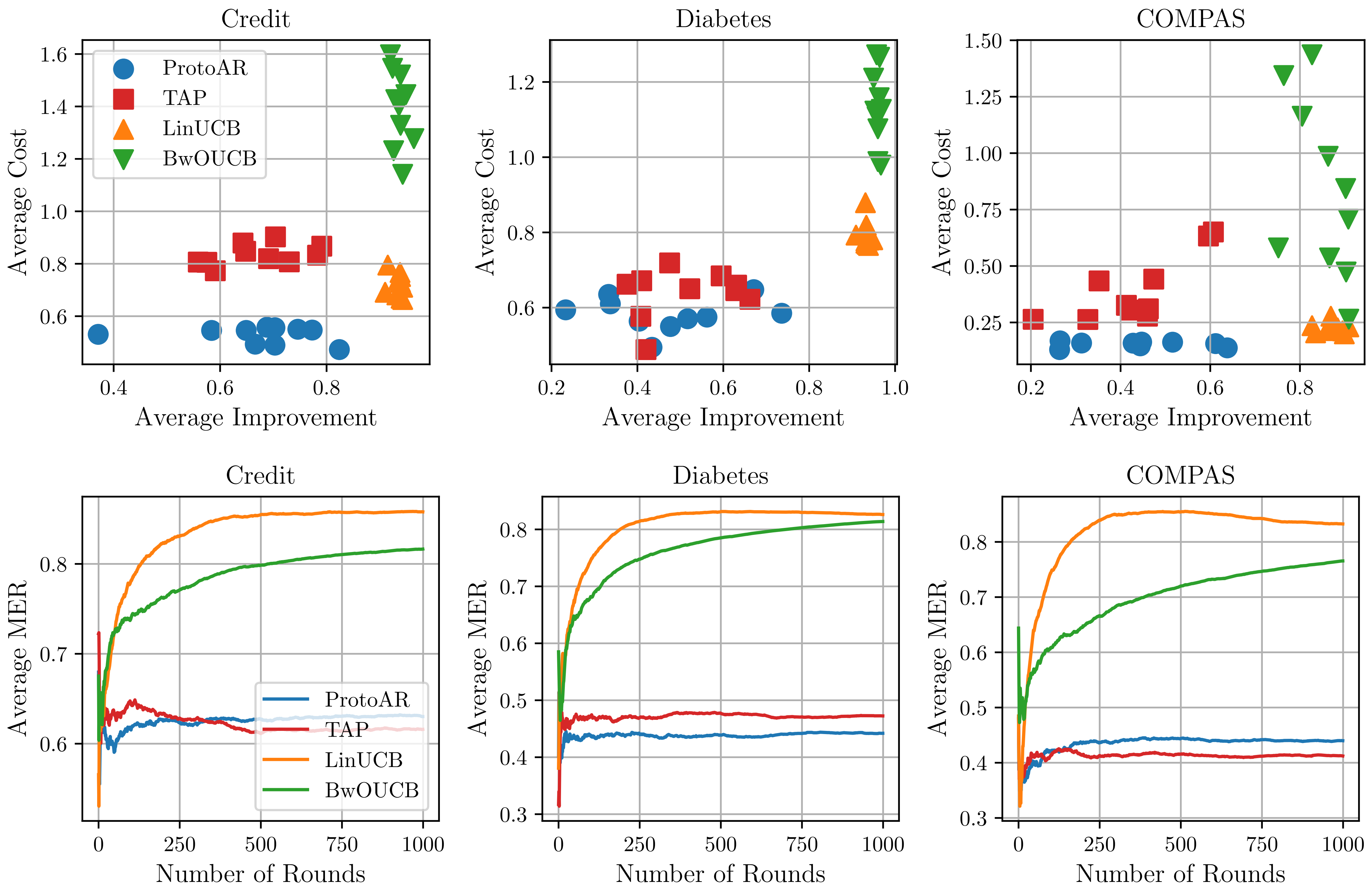


Experiments

Achieve higher improvement than baselines without significantly degrading cost

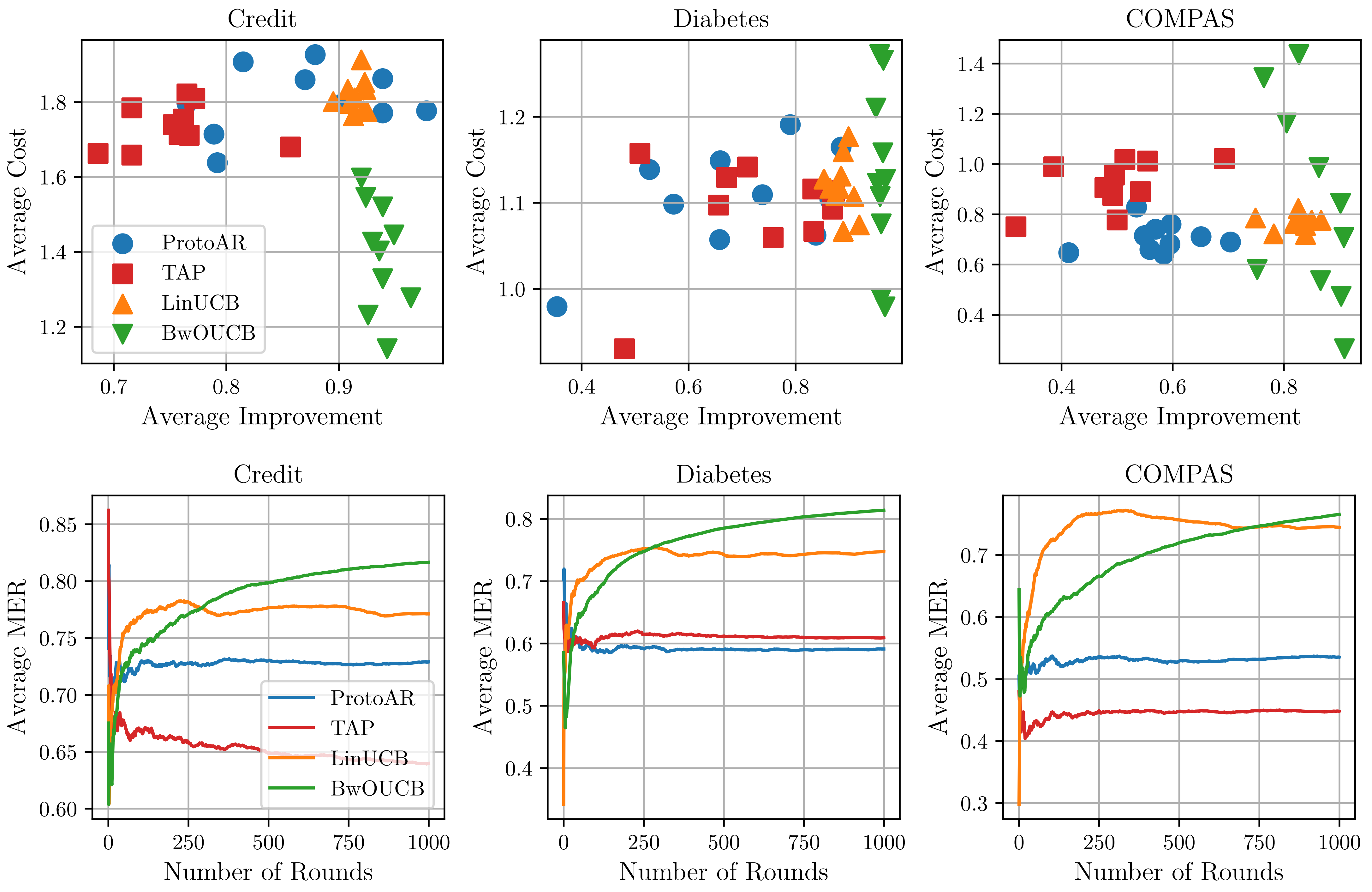
Case 1. "Noiseless" Cost Scenario

- Our LinUCB attained higher improvements while maintaining comparable costs



Case 2. "Noisy" Cost Scenario

- The performance of our BwOUCB was better than or close to others in many cases



Summary

Provide improvement-oriented recourse actions from the long-term perspective

- Introduce a new online learning task: *algorithmic recourse for long-term improvement*
- Propose two algorithms based on the *contextual linear bandit* and *Bayesian optimization*
- Demonstrate that our methods could provide actions for improving the real-world outcome