REGULATING HUMAN-IN-THE-LOOP LEARNING FOR DYNAMIC LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language model (LLM) platforms increasingly host multiple models whose response quality evolves through user interactions and online adaptation. This creates a human-in-the-loop model-selection problem in which myopic users favor the best immediate option, while the platform seeks to maximize long-run user satisfaction. We formulate this problem as a parametric stochastic rising bandit with myopic users, using saturation families to capture adaptation-driven reward growth. For a platform-controlled benchmark, we propose GAP-UCB, a growth-aware prospective UCB algorithm that fits reward trajectories via nonlinear least squares and compares models using optimistic prospective rewards. GAP-UCB achieves $O(\sqrt{T \log T})$ regret, improving over the existing $O(T^{2/3})$ guarantee for general stochastic rising rested bandits. We then return to the user-driven setting and develop iGAP-UCB, an ex-post budget-balanced payment mechanism that regulates users toward the platform-preferred action while preserving the same regret order up to a fee-dependent factor. Experiments on real LLM fine-tuning trajectories validate the effectiveness of GAP-UCB in the platform-controlled setting and show that iGAP-UCB closely tracks the platform-preferred benchmark under budget-balanced user regulation.

1 INTRODUCTION

Large language models (LLMs) are now widely deployed in applications such as text summarization Cao et al. (2024), mathematical reasoning Gao et al. (2023), and multimodal understanding Yin et al. (2024). Meanwhile, LLM platforms such as OpenAI OpenAI (2026), Together AI Together AI (2026), and Hugging Face Hugging Face (2026) increasingly host multiple candidate models with different capabilities. When a user arrives with a query, the platform publishes its current performance estimates for the available models. Based on these estimates, the user selects the LLM expected to deliver the highest immediate response quality and interacts with it. The resulting feedback allows the platform to update its estimates and support better model selection for subsequent users. This repeated cycle constitutes a human-in-the-loop online model selection problem whose objective is to maximize long-run user satisfaction, as illustrated in Figure 1.

Unlike standard human-in-the-loop model selection, feedback in an LLM system plays a dual role by revealing the selected model's current quality while also improving the model itself. Specifically, interaction data can be used to adapt the selected LLM through fine-tuning, lightweight parameter updates, or retrieval-based mechanisms Sun et al. (2025); Wang et al. (2025); Kaplan et al. (2020a); Poon et al. (2026); Han et al. (2024); Kuramoto & Suzuki (2025). As a result, model performance is dynamic rather than fixed, since repeated selection may improve a model's future response quality. Figure 2 illustrates this phenomenon on real LoRA-based fine-tuning trajectories on GSM8K Cobbe et al. (2021). Different LLMs exhibit clear increasing-then-saturating response-quality curves, but with heterogeneous initial qualities, growth rates, and saturation levels. The improvement can be substantial (up to 50% in absolute accuracy), and even model rankings may change over time. For example, Yi-6B is initially the best model, but is eventually overtaken by Qwen2.5-1.5B and Llama-3.2-3B. Thus, selecting an LLM does not merely reveal its current quality, but can also improve its future quality through continued adaptation.

Bandit models provide a powerful framework for studying human-in-the-loop learning through the exploration-exploitation tradeoff. Traditional stationary bandits treat exploration primarily as

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

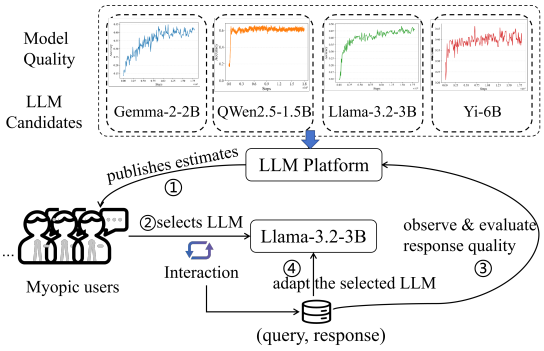


Figure 1: Human-in-the-loop online LLM selection on a platform. The platform publishes current performance estimates, the arriving user selects and interacts with an LLM, and the resulting interaction data is used both for online learning and for further adaptation of the selected model.

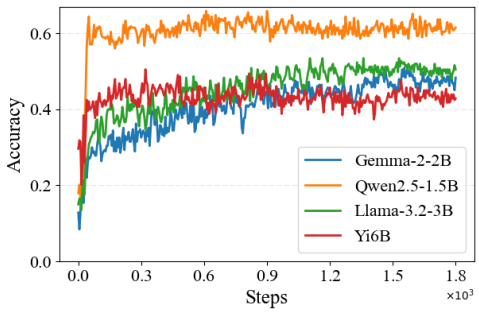


Figure 2: LoRA-based adaptation trajectories of four open-source LLMs on GSM8K, with response quality measured by accuracy over adaptation steps. The trajectories increase and then saturate, with different initial qualities, growth rates, and saturation levels.

a means of reducing uncertainty, whereas in our setting, exploration also generates interactions that improve future model quality, making the tradeoff substantially more challenging. Consequently, policies such as UCB Auer (2002) and Thompson sampling Agrawal & Goyal (2012) can be shortsighted. Rising bandits provide a natural model for such evolving rewards, and existing methods achieve $O(T^{2/3})$-type regret guarantees under broad monotonicity and concavity assumptions. However, these general methods do not exploit a key empirical feature of LLM adaptation, i.e., reward trajectories often follow simple increasing-and-saturating patterns. As shown in Figure 2 and further quantified in Appendix B.2, such trajectories can be well approximated by low-dimensional saturation families. This additional structure suggests that more accurate long-horizon prediction and sharper regret guarantees may be possible.

A further challenge is that users are myopic, seeking to maximize their immediate rewards with no incentive to explore and improve models for the platform's long-term benefit. Because the platform is not the final decision-maker, users naturally choose the currently best model, limiting the exploration of underexplored but high-potential alternatives and depriving them of the interactions needed for future improvement. This self-reinforcing effect can lock users into selecting a suboptimal model, leading to high long-run regret. Hence, accurate learning alone is insufficient; the platform must also regulate user decisions while avoiding persistent subsidy.

To address these two challenges, we formulate online LLM selection as a parametric stochastic rising bandit with myopic users, where the reward represents the response quality delivered to the arriving user and hence the user's realized satisfaction. The formulation captures the coupled difficulties of learning under adaptation-driven reward growth and regulating myopic user choice. To isolate the first difficulty, we first study a benchmark in which the platform directly controls model selection and design a growth-aware learning rule that fits each LLM's adaptation trajectory using a finite library of saturation families and compares candidate LLMs at a common prospective adaptation state. We then return to the user-driven setting and develop a payment mechanism that regulates myopic users toward the platform-preferred action while ensuring ex post budget balance, thereby allowing the platform to learn the optimal model without incurring a long-term deficit. Experiments on real-world LLM selection tasks complement our theoretical results and demonstrate the strong practical applicability of our methods. Our main contributions are summarized as follows:

- In Section 2, we formulate human-in-the-loop online LLM selection with adaptation driven performance growth as a parametric stochastic rising bandit with myopic users, capturing the joint learning-and-regulation challenge on LLM platforms.
- We propose GAP-UCB in Section 3, a growth-aware prospective UCB algorithm for a platform-controlled benchmark. Under the parametric formulation, GAP-UCB achieves an $O(\sqrt{T \log T})$ regret bound, improving over the $O(T^{2/3})$ guarantee for general stochastic rising rested bandits.

- In Section 4, we develop iGAP-UCB, an ex post budget-balanced payment mechanism that regulates myopic user decisions toward the platform-preferred action while ensuring incentive compatibility. The mechanism preserves the same regret order as GAP-UCB up to a factor determined by the fee cap p.
- In Section 5, we evaluate the proposed framework on real LLM fine-tuning trajectories and validate the effectiveness of GAP-UCB in the platform-controlled setting and show that iGAP-UCB closely tracks GAP-UCB under budget-balanced user regulation.

A discussion of related work and all proofs are provided in the appendices.

2 Problem Formulation

We formulate LLM users' online model selection as a stochastic rising rested bandit with myopic users. This section specifies the system model, the dynamic reward formulation, the platform objective, and the user-side decision model.

2.1 System Model

Consider an LLM platform with K candidate models, indexed by $[K] := \{1, 2, \ldots, K\}$. Time proceeds in discrete rounds $t = 1, 2, \ldots, T$, each corresponding to one arriving user query. As illustrated in Figure 1, in each round the platform first publishes its current performance estimates, the arriving user then selects a model $I_t \in [K]$ and interacts with it, and the platform evaluates the response quality and observes the scalar reward y_t delivered to the arriving user:

$$y_t = r_{I_t}(N_{I_t,t}) + \eta_t, \tag{1}$$

where $N_{i,t}$ denotes the number of times model i has been selected before round t, $r_i(n) \in (0, 1]$ is its expected response quality after n completed adaptations, and η_t is independent and identically distributed (*i.i.d.*) σ-sub-Gaussian noise over t. After observing the reward, the selected model is adapted once, so $N_{i,t+1} = N_{i,t} + \mathbf{1}\{i = I_t\}$ for all $i \in [K]$. We also let $\tilde{y}_{i,n} := r_i(n) + \eta_{i,n}$ denote the n-th model-wise observation, where $\eta_{i,n}$ is the corresponding re-indexed noise term from Eq. (1). In LLM applications, y_t can be computed using automatic evaluation tools such as correctness verification Cobbe et al. (2021), BERTScore Zhang et al. (2020), or LLM-based judges Zheng et al. (2023). Query-level variability is absorbed into the observation noise.

2.2 Parametric Dynamic Reward Model

A key feature of our setting is that model quality evolves through repeated adaptation. As discussed in Section 1, empirical adaptation trajectories generally exhibit increasing-then-saturating behavior, while their growth patterns vary across models. We therefore model each LLM's reward trajectory using a finite library of low-dimensional saturation families.

Let $\mathcal{M} := [M]$ index the candidate families. For each $m \in \mathcal{M}$, let $f_m(n; \theta)$ denote the improvement function with parameter vector $\theta \in \Theta_m$, where Θ_m is a compact parameter space. The complete family library is $\mathcal{F} := \{f_m(\cdot; \theta) \mid m \in \mathcal{M},\ \theta \in \Theta_m\}$. For each LLM i, there is an unknown family identity $m_i^* \in \mathcal{M}$ and an unknown parameter vector $\theta_i^* \in \Theta_{m_i^*}$ such that its improvement curve is $f_{m_i^*}(n; \theta_i^*)$. Its expected reward after n adaptations can then be written as $r_i(n) = \mu_i + f_{m_i^*}(n; \theta_i^*)$, where μ_i denotes the baseline reward before online adaptation.

In this paper, we instantiate the family library using three representative increasing-and-saturating functions: exponential, polynomial, and hyperbolic saturation for LLM reward modeling. For all three families, the improvement function is parameterized by $\theta = (c, a)$, where $c > 0$ represents the asymptotic improvement magnitude, and $a > 0$ controls the adaptation speed under different families. Each family can be written in the unified form

$$f_m(n; (c, a)) = c\big(1 - h_{m,a}(n)\big), \tag{2}$$

where

$$h_{m,a}(n) = \begin{cases} e^{-an}, & m = \text{exponential saturation}, \\ (n+1)^{-a}, & m = \text{polynomial saturation}, \\ (an+1)^{-1}, & m = \text{hyperbolic saturation}. \end{cases} \tag{3}$$

In each case, $h_{m,a}(n)$ is nonnegative, decreases with n, and converges to zero, so $f_m(n;(c,a))$ increases toward the saturation level c. The exponential family approaches its asymptote exponentially fast, the polynomial family allows heavier-tailed convergence, and the hyperbolic family provides a rational-decay alternative. Throughout the paper, for each $m \in \mathcal{M}$, we restrict the parameter vector θ to a rectangular domain $\Theta_m = [c_{m,\min}, c_{m,\max}] \times [a_{m,\min}, a_{m,\max}]$, where the parameter domains are bounded away from zero.

The three families are representative low-dimensional models of increasing-with-diminishing-returns adaptation dynamics and are consistent with LLM learning-curves considered in prior work (Kaplan et al., 2020b; Yuan et al., 2023; Zhang et al., 2024). Related work on rising bandits and online LLM selection similarly characterizes reward evolution using these families Metelli et al. (2022); Xia et al. (2024). We conduct a fitting analysis on real LLM adaptation trajectories to show that the three parameterized families capture the main evolution patterns of adaptation-driven response quality, with Table 2 in Appendix B.2 reporting the RMSE and R^2 values for the trajectories shown in Figure 2, thereby motivating our focus on these representative families.

Throughout this paper, we assume that for each LLM, its expected improvement trajectory belongs to the finite candidate library $\mathcal{F}$. The three families above are our concrete instantiation, and the analysis applies to any finite library satisfying the regularity conditions used later.

2.3 Platform Regret and Myopic User Choice

Platform Objective. The platform aims to maximize cumulative expected user satisfaction over a horizon of T rounds, as measured by the cumulative expected response quality $\sum_{t=1}^{T} r_{I_t}(N_{I_t,t})$.

As a benchmark, consider a full-control oracle that knows all reward functions and directly controls model selection. For stochastic rising bandits, the optimal oracle policy repeatedly selects the single model $i^* \in \arg\max_{i\in[K]} \sum_{t=1}^{T} r_i(t-1)$ (Metelli et al., 2022, Theorem 4.1), which is unknown in advance. We then evaluate a policy π by its cumulative expected regret relative to this oracle:

$$R_\pi(T) := \mathbb{E}\left[\sum_{t=1}^{T} r_{i^*}(t-1) - \sum_{t=1}^{T} r_{I_t}(N_{I_t,t})\right]. \tag{4}$$

Equivalently, the platform's objective is to minimize $R_\pi(T)$.

User Decision Rule. In practice, the serving model is selected not by the platform but by the arriving user based on the platform's published estimates. Let $\widehat{r}_{i,t}(\cdot)$ denote the platform's estimated reward trajectory of model i before round t. Given the current adaptation states $\{N_{i,t}\}_{i=1}^{K}$, a myopic user prefers the model with the highest immediate estimated utility:

$$j_t \in \operatorname*{argmax}_{i\in[K]} \widehat{r}_{i,t}(N_{i,t}). \tag{5}$$

This user-side decision rule is misaligned with the platform's long-term objective. An underexplored model may have lower immediate reward even when its long-run value is higher, so myopic users tend not to select it. As a result, this model receives fewer interactions, adapts more slowly, and becomes even less attractive in the short run. Regulating users' selections is therefore essential in our setting. Later, in Section 4, we show that, without such regulation, this feedback loop can lead to linear regret, thereby motivating a payment mechanism that regulates users toward exploratory actions.

3 Platform-Preferred Benchmark: GAP-UCB

We first study a benchmark setting in which the platform directly controls model selection. This isolates the learning challenge caused by adaptation-driven reward dynamics, before we return to user-side decision-making and regulation mechanisms in Section 4.

3.1 Main Ideas

A natural baseline for platform-controlled online LLM selection is to follow the upper confidence bound (UCB) principle and select the model with the largest optimistic estimate of its immediate

reward Lattimore & Szepesvári (2020). However, this can be fundamentally short-sighted in rising bandits because different LLMs are evaluated at different adaptation states $N_{i,t}$. We explicitly show in Proposition C.1 of Appendix C.1 that this strategy can incur linear regret.

To compare models fairly, we instead evaluate them at a common prospective adaptation state, following the key idea used in prior rising-bandit methods Metelli et al. (2022). Existing methods typically estimate future rewards through local linear extrapolation, which is deliberately optimistic under concavity but can be conservative over a long horizon when rewards exhibit strong diminishing returns. In our setting, LLM adaptation trajectories are well captured by low-dimensional increasing-and-saturating families, allowing us to estimate long-run rewards directly from parametric trajectory models. Since different LLMs may follow different growth patterns, we fit all candidate families and use the best-fitting one for each model.

3.2 The GAP-UCB Algorithm

We now combine online trajectory fitting with prospective optimism to construct the growth-aware prospective UCB (GAP-UCB) algorithm (Algorithm 1). At the beginning of round $t > K$, model i

Algorithm 1 Growth-Aware Prospective UCB (GAP-UCB)

Input: K, $\{\mu_i\}_{i=1}^K$, family library $\mathcal{F} = \{f_m(\cdot;\theta) : m \in \mathcal{M}, \theta \in \Theta_m\}$, confidence level δ
Initialize: $N_{i,1} \leftarrow 0$ for every $i \in [K]$
for $t = 1, \ldots, K$ **do**
    Select LLM $I_t \leftarrow t$, observe y_t, and store $\tilde{y}_{I_t,0} \leftarrow y_t$
    Update $N_{i,t+1} \leftarrow N_{i,t} + \mathbf{1}\{i = I_t\}$ for all $i \in [K]$
end for
for $t = K+1, \ldots, T$ **do**
    for each $i \in [K]$ **do**
        Compute $(\widehat{m}_{i,N_{i,t}}, \widehat{\theta}_{i,N_{i,t}})$ by the joint NLS estimator in Eq. (6)
        Construct $\widehat{r}_{i,t}(\cdot)$ by Eq. (7)
    end for
    Select I_t by Eq. (8)
    Observe $y_t = r_{I_t}(N_{I_t,t}) + \eta_t$ and set $\tilde{y}_{I_t,N_{I_t,t}} \leftarrow y_t$
    Update $N_{i,t+1} \leftarrow N_{i,t} + \mathbf{1}\{i = I_t\}$ for all $i \in [K]$
end for

has been selected $n = N_{i,t}$ times with observations $\{(s, \widetilde{y}_{i,s})\}_{s=0}^{n-1}$. Based on the observations, the platform jointly estimates the family identity and its parameters via nonlinear least squares (NLS):

$$(\widehat{m}_{i,n}, \widehat{\theta}_{i,n}) \in \arg\min_{m \in \mathcal{M}, \theta \in \Theta_m} \sum_{s=0}^{n-1} \left[\widetilde{y}_{i,s} - \mu_i - f_m(s;\theta)\right]^2 . \tag{6}$$

Since the family set $\mathcal{M}$ is finite, Eq. (6) can be implemented by fitting each family separately and selecting the one with the smallest residual error. Then, the estimated reward trajectory is

$$\widehat{r}_{i,t}(\cdot) := \mu_i + f_{\widehat{m}_{i,n}}(\cdot; \widehat{\theta}_{i,n}). \tag{7}$$

A key feature of GAP-UCB is that candidate models are not compared at their current adaptation states $N_{i,t}$, which may differ substantially across models. Instead, GAP-UCB evaluates all models at the common prospective index $t-1$ and selects

$$I_t \in \arg\max_{i \in [K]} \{\widehat{r}_{i,t}(t-1) + \beta_{i,t}(\delta)\} . \tag{8}$$

The common prospective index $t-1$ places all models at the same hypothetical adaptation state, removing the bias induced by unequal past selection counts. The confidence radius $\beta_{i,t}(\delta)$ controls the prediction uncertainty of the jointly fitted trajectory over the finite candidate library.

3.3 Regret Analysis

We next establish a prediction-confidence bound for the joint estimator in Eq. (6) and use it to analyze the regret of GAP-UCB. Importantly, the analysis does not require GAP-UCB to correctly identify the latent family label m_i^* at every round. Instead, it directly controls the prediction error of the fitted curve selected by the joint estimator in Eq. (6).

Proposition 3.1 (Joint-Family Reward Concentration). *Let* $b_s := \sup_{m\in\mathcal{M},a\in\Theta_{a,m}} h_{m,a}(s)$, $u_n := 2c_{\max}\sqrt{\frac{1}{n}\sum_{s=0}^{n-1} b_s^2}$, *and* $c_{\max} = \max_m c_{m,\max}$. *For any confidence level* $\delta \in (0,1)$, *define*

$$A_n(\delta) := 2\sigma\sqrt{\frac{2\log(Mn) + C_0 + 2\log(4/\delta)}{n}} \quad \textit{and} \quad B_n(\delta) := M_J\left(\frac{A_n(\delta)}{\sqrt{n}} + \frac{2\sigma\sqrt{2\log(2/\delta)}}{n}\right)$$

where C_0, M_J *are two constants. Consider the joint NLS estimator* $(\widehat{m}_{i,n}, \widehat{\theta}_{i,n})$ *in Eq.* (6), *and let* $\widehat{r}_i^{(n)}(\cdot) := \mu_i + f_{\widehat{m}_{i,n}}(\cdot; \widehat{\theta}_{i,n})$ *denote the corresponding* n*-sample fitted trajectory. Then, for every* $s \geq 0$, *with probability at least* $1-\delta$,

$$|\widehat{r}_i^{(n)}(s) - r_i(s)| \leq u_n + \max\left\{2A_n(\delta), 2\sqrt{B_n(\delta)}\right\} + 2c_{\max}b_s.$$

Proposition 3.1 immediately suggests the confidence radius used in GAP-UCB, which is

$$\beta_{i,t}(\delta) := u_{N_{i,t}} + \max\left\{2A_{N_{i,t}}(\delta), 2\sqrt{B_{N_{i,t}}(\delta)}\right\} + 2c_{\max}b_{t-1}. \tag{9}$$

Note that prediction uncertainty is not the only source of regret. Even with perfect parameter knowledge, pulling the wrong model early in a growing-reward environment can incur a persistent loss by delaying future improvement. We capture this structural effect through the following quantity.

Definition 3.2 (Tail-Sum Growth Function). For a candidate curve $f_m(\cdot;\theta)$, define its increment $g_{m,\theta}(n) := f_m(n;\theta) - f_m(n-1;\theta)$ and its tail-sum growth function

$$G_{m,\theta}(N) := \sum_{n=0}^{N-1}\sum_{s=n+1}^{\infty} g_{m,\theta}(s).$$

This quantity measures the cumulative future reward loss incurred when model i is selected only N times instead of from the beginning. Let $G_{\max}$ be a concave, non-decreasing benchmark such that $G_{m,\theta}(N) \leq CG_{\max}(N)$ for every candidate family and admissible parameter.

Theorem 3.3 (Regret Bound of GAP-UCB). *Under the conditions of Proposition 3.1, let GAP-UCB use the confidence radius* $\beta_{i,t}(\delta)$ *in Eq.* (9) *with* $\delta = 1/(KT^2)$. *Then, for* $T \geq K$,

$$R_{\text{GAP-UCB}}(T) \leq O\left(\sqrt{KT\log(MKT)}\right) + \Gamma(T),$$

where

$$\Gamma(T) := KC\,G_{\max}(T/K) + 2\sum_{t=K+1}^{T} u_{N_{I_t,t}} + 4c_{\max}\sum_{t=K+1}^{T} b_{t-1}.$$

Remark 3.4. *The bound in Theorem 3.3 separates the statistical learning cost from the growth-dependent cost. The first term* $O(\sqrt{KT(\log(MKT))}$ *captures the cost of learning reward trajectories over the finite family library.* $KC\,G_{\max}(T/K)$ *is the structural loss caused by delayed adaptation in a growing-reward environment. The remaining two terms account for finite-sample trajectory estimation and transient disagreement among admissible saturation families. Thus, the analysis directly controls the prediction error of the selected trajectory and does not require consistent recovery of the latent family identity.*

We next instantiate the $\Gamma(T)$ for the representative reward families introduced in Section 2.2.

Corollary 3.5 (Exponential–Polynomial–Hyperbolic Library). *Suppose each LLM follows one of the exponential, polynomial, or hyperbolic saturation families in Section 2.2, and the corresponding true trajectory is contained in* $\mathcal{F}$. *Then the regret in Theorem 3.3 becomes*

$$R_{\text{GAP-UCB}}(T) = \begin{cases} O\left(\sqrt{KT\log(MKT)} + K\big(1+\log(T/K)\big) + \log T\right), & a_{\text{poly,min}} \geq 1, \\ O\left(\sqrt{KT\log(MKT)} + K^{a_{\text{poly,min}}}T^{1-a_{\text{poly,min}}}\right), & a_{\text{poly,min}} \in (0,1). \end{cases}$$

In particular, exponential and hyperbolic saturation contribute only logarithmic growth-dependent terms. The polynomial family can converge more slowly, yielding the additional $T^{1-a_{\text{poly,min}}}$ term when $a_{\text{poly,min}} < 1$.

Remark 3.6. *Compared with the* $O(KT^{2/3})$ *regret guarantee for general stochastic rising rested bandits Metelli et al. (2022), the parametric structure exploited by GAP-UCB yields a leading*

$O(\sqrt{KT \log T})$ *statistical term. It also improves the dependence on reward growth for representative saturation families. For example, under hyperbolic saturation, the corresponding shape-dependent term in Metelli et al. (2022) can scale linearly with T, whereas our instantiated bound grows only logarithmically.*

4 Regulating User Decisions: iGAP-UCB

Section 3 characterized the platform-preferred policy under direct control. We now return to the actual user-decision setting, in which the arriving user makes the final model choice. The challenge is that, even with accurate reward estimates, myopic users optimize only immediate utility and may therefore deviate from the platform-preferred long-term action. This section develops a payment mechanism that regulates users toward the GAP-UCB action while maintaining ex-post budget balance. In this section, we first show why regulation is necessary, then present the iGAP-UCB mechanism, and finally establish its regret guarantee.

4.1 Why Regulation is Necessary

Even with perfect knowledge of the reward trajectories, myopic user decisions can remain misaligned with the platform's long-run objective. An underadapted model with high long-run value may offer lower immediate reward and therefore receive too few selections to realize its future potential. This feedback can persist over time and lead to linear regret.

Proposition 4.1 (Linear Regret Without Regulation)**.** *Even if $\widehat{r}_{i,t} = r_i$ for all $i \in [K]$, the myopic policy in Eq.* (5) *without regulation can incur linear regret, i.e., $R_{\mathrm{myopic}}(T) = \Omega(T)$.*

Proposition 4.1 shows that accurate learning alone is insufficient in the user-driven setting. To achieve the platform's long-term objective, the platform must additionally regulate users' decisions.

4.2 Mechanism Design to Regulate Decisions

To align myopic user decisions with the platform's long-run objective, we build a payment mechanism on top of GAP-UCB. At round $t > K$, let i_t denote the model recommended by GAP-UCB in Eq. (8). The arriving myopic user instead prefers the model j_t in Eq. (5), which maximizes the immediate estimated reward. If $i_t = j_t$, no intervention is needed. Otherwise, to regulate the user toward i_t, the platform must compensate the user's immediate utility gap relative to j_t, namely

$$c_t := \widehat{r}_{j_t,t}(N_{j_t,t}) - \widehat{r}_{i_t,t}(N_{i_t,t}). \tag{10}$$

Repeatedly paying c_t from external funds, however, can lead to an unbounded platform deficit. We therefore require the mechanism to be *ex-post budget balance* Börgers (2015), ensuring that the cumulative budget remains nonnegative at every round. Let $B_{t-1} \geq 0$ denote the available budget before round t. The platform charges a model-independent fee

$$p_t := \min\left\{p, \widehat{r}_{j_t,t}(N_{j_t,t})\right\}. \tag{11}$$

where $p \in (0, 1]$ is a fee cap. The truncation ensures that even the user's preferred model retains nonnegative estimated utility after paying the fee, while the collected fee contributes to the compensation budget. Then, the platform offers compensation only when the available budget after collecting p_t is sufficient. This leads to the following ex-post budget-balanced mechanism.

Definition 4.2 (iGAP-UCB Mechanism)**.** Given history $H_{t-1} := \{(I_s, y_s)\}_{s=1}^{t-1}$ and budget B_{t-1}, the mechanism charges the fee p_t in Eq. (11) and uses the compensation rule

$$x_t(i) = \begin{cases} c_t\, \mathbf{1}\{i = i_t\}, & \text{if } B_{t-1} + p_t \geq c_t, \\ 0, & \text{otherwise.} \end{cases}$$

The user's estimated net utility from selecting model i then becomes

$$u_i(t) := \widehat{r}_{i,t}(N_{i,t}) + x_t(i) - p_t. \tag{12}$$

Note that p_t is independent of the selected model and thus does not affect the user's ranking of the models. The user then selects $I_t \in \arg\max_{i \in [K]} u_i(t)$ with ties broken in favor of i_t. Therefore, whenever $B_{t-1} + p_t \geq c_t$, the mechanism implements the platform-preferred action. After the user selects I_t, the budget is updated as $B_t = B_{t-1} + p_t - x_t(I_t)$.

The complete procedure is summarized in Algorithm 2 of Appendix D. We formalize the three important guarantees of our iGAP-UCB mechanism in the next lemma.

Lemma 4.3. *For every round $t > K$, iGAP-UCB satisfies:*

(i) Ex-post budget balance: $B_t \geq 0$.

(ii) Action-level implementability: *whenever $B_{t-1} + p_t \geq c_t$, the platform-preferred model i_t is utility-maximizing for the user under tie-breaking in favor of i_t in Eq.* (8).

(iii) Individual rationality: *the user's realized net utility is nonnegative, i.e., $u_{I_t}(t) \geq 0$.*

4.3 Regret Guarantee

We now characterize the regret of iGAP-UCB. The only difference between iGAP-UCB and the platform-controlled GAP-UCB benchmark arises in budget-deficit rounds, when the available budget is insufficient to implement the GAP-UCB recommendation. Hence, the regret of iGAP-UCB can be decomposed into the regret of the underlying GAP-UCB policy and the additional regret caused by such deviations.

Theorem 4.4 (Regret of iGAP-UCB)**.** *Under the same conditions and confidence radius as in Theorem 3.3, let $\delta = 1/(KT^2)$. For $T \geq K$ and the fee cap $p \in (0, 1]$,*

$$R_{\text{iGAP-UCB}}(T) = O\left((1 + p^{-1})\left(\sqrt{KT\log(MKT)} + \Gamma(T)\right)\right),$$

where $\Gamma(T)$ is the same family-dependent term as in Theorem 3.3.

Theorem 4.4 shows that iGAP-UCB preserves the asymptotic regret order of GAP-UCB for any fixed $p > 0$. The multiplicative factor $(1 + p^{-1})$ quantifies the additional cost of budget-deficit rounds. That is, larger p allows the budget to accumulate faster, reducing the number of rounds in which the GAP-UCB recommendation cannot be implemented.

5 Experiments

Models and fine-tuning task. We consider four representative open-source LLMs from Hugging Face Wolf et al. (2020) as candidate models: Gemma-2-2B Team (2024), Qwen2.5-1.5B Team et al. (2024), Llama-3.2-3B AI@Meta (2024), and Yi-6B Young et al. (2024). These models have moderate base performance on mathematical reasoning, leaving sufficient room for adaptation-driven improvement. To capture performance evolution over adaptation steps, we fine-tune each model with LoRA on GSM8K Cobbe et al. (2021), a benchmark of multi-step mathematical word problems. From the 7,473 training samples, we use 7,200 for supervised fine-tuning and reserve the remaining 273 for out-of-sample evaluation. Fine-tuning is conducted on an NVIDIA RTX 5090 GPU (32GB VRAM) with an effective batch size of 16. We use model-specific learning rates and otherwise consistent LoRA configurations across models. Each model is trained for 4 epochs, resulting in 1,800 adaptation steps. Full training details are provided in Appendix E.1.

Setup. We simulate online LLM selection over $T = 1800$ rounds using the empirical fine-tuning trajectories of the $K = 4$ LLMs. GAP-UCB uses a candidate library of $M = 3$ saturation families—exponential, polynomial, and hyperbolic—and selects the best-fitting family for each LLM from the observations available at each round. We set the $\sigma = 0.1$ and $\delta = 1/(KT^2)$ in the confidence radius. Additional implementation details are provided in Appendix E.1.

Baselines. We compare against the following methods:

- **RedUCB** Metelli et al. (2022): a rising-bandit baseline based on local linear extrapolation with a sliding-window estimator;
- **TI-UCB** Xia et al. (2024): an LLM-selection baseline that approximates the increasing reward trend using piecewise linear fitting before a detected convergence point;
- **Greedy**: always selects the model with the largest estimated immediate reward, without accounting for long-term growth.

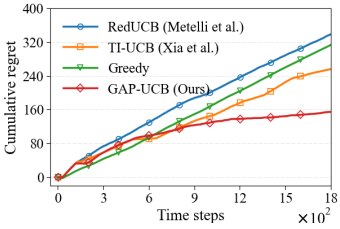
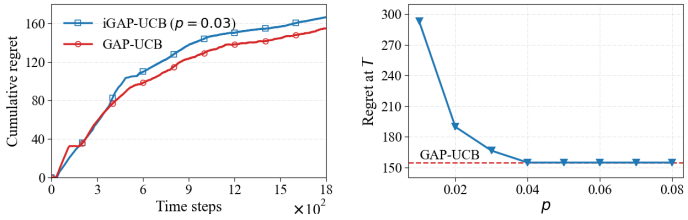


Figure 3: Regret in the platform-controlled benchmark.



(a) iGAP-UCB vs. GAP-UCB (b) Final regret under different p

Figure 4: Performance of iGAP-UCB in the user-driven setting

The experiments are designed to answer the following questions: **Q1:** Does GAP-UCB outperform existing rising-bandit baselines in the platform-controlled case? **Q2:** How closely can iGAP-UCB track the platform-controlled GAP-UCB benchmark under myopic user decisions? **Q3:** How does the per-round fee cap p affect the regret of iGAP-UCB?

Platform-controlled benchmark. We first evaluate GAP-UCB in the platform-controlled benchmark. Figure 3 shows the cumulative regret over $T = 1800$ rounds. GAP-UCB consistently achieves the lowest long-run regret among the compared methods. Greedy performs competitively in early rounds because it exploits the model with the highest immediate estimated reward, which is initially Yi-6B. However, it ignores future reward growth and therefore over-commits to a model that is strong initially but suboptimal in the long run. RedUCB incurs substantially higher regret than Greedy in this experiment. The reason is that it explores all models too aggressively and does not concentrate quickly enough on the long-term optimal LLM within the finite horizon $T = 1800$. TI-UCB performs better than RedUCB, but remains inferior to GAP-UCB because its piecewise-linear approximation does not fully capture the global increasing-and-saturating pattern of the adaptation trajectories. The final model-selection counts are reported in Appendix E.2.

We further examine the role of adaptive family selection in Appendix E.3. The ablation results show that selecting the best-fitting family online yields substantially lower regret than fixing any single saturation family throughout the horizon.

User-driven setting. We next evaluate iGAP-UCB under myopic user decisions. Figure 4a compares iGAP-UCB (with $p = 0.03$) against the platform-controlled GAP-UCB. Despite user-side decision-making, iGAP-UCB closely tracks GAP-UCB, with the gap arising from rounds in which the available budget is insufficient to implement the platform-preferred action. This empirically supports the regret result in Theorem 4.4. (Q2). Figure 4b shows the effect of the fee cap p. A larger p allows the budget to accumulate more quickly, reducing the frequency of budget-deficit rounds and enabling iGAP-UCB to implement the GAP-UCB recommendation more often. Consequently, the final regret decreases and approaches the platform-controlled benchmark as p increases. (Q3).

Additional experiments in Appendix E.2 examine how the same budget-balanced mechanism interacts with alternative platform-controlled baseline policies.

6 Conclusion

We studied human-in-the-loop online LLM selection when repeated model usage improves future response quality while model choices are made by myopic users. We formulated this problem as a parametric stochastic rising bandit with myopic users, proposed GAP-UCB to characterize the platform-preferred benchmark under direct control, and developed iGAP-UCB to regulate user decisions under ex-post budget balance. Under the parametric saturation model, GAP-UCB achieves an $O(\sqrt{T \log T})$ regret bound, while iGAP-UCB preserves the same asymptotic regret order up to a fee-dependent factor. Experiments on real LLM fine-tuning trajectories demonstrate the advantage of GAP-UCB over existing platform-controlled baselines and show that iGAP-UCB closely tracks the platform-controlled benchmark while exhibiting the predicted fee-performance tradeoff.

AI USE STATEMENT

Generative AI tools were used to assist in refining theoretical arguments and checking mathematical proofs, providing feedback on experimental methodology and interpretation of results, and developing and debugging experimental code. They were additionally used for manuscript organization, clarity, and language editing. All AI-assisted mathematical claims and proofs were independently checked by the authors, and all AI-assisted code and experimental procedures were reviewed and tested by the authors. The authors take full responsibility for the final content of the paper.

REFERENCES

Pranjal Aggarwal, Aman Madaan, Ankit Anand, Srividya Pranavi Potharaju, Swaroop Mishra, Pei Zhou, Aditya Gupta, Dheeraj Rajagopal, Karthik Kappaganthu, Yiming Yang, Shyam Upadhyay, Manaal Faruqui, and Mausam . Automix: Automatically mixing language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Shipra Agrawal and Navin Goyal. Analysis of thompson sampling for the multi-armed bandit problem. In Shie Mannor, Nathan Srebro, and Robert C. Williamson (eds.), *Proceedings of the 25th Annual Conference on Learning Theory*, volume 23 of *Proceedings of Machine Learning Research*, pp. 39.1–39.26, Edinburgh, Scotland, 25–27 Jun 2012. PMLR. URL https://proceedings.mlr.press/v23/agrawal12.html.

AI@Meta. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Baran Atalar. Neural bandit based optimal llm selection for pipeline of tasks. *ACM SIGMETRICS Performance Evaluation Review*, 53(3):15–17, 2026.

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

Tilman Börgers. *An introduction to the theory of mechanism design*. Oxford university press, 2015.

Tianyu Cao, Natraj Raman, Danial Dervovic, and Chenhao Tan. Characterizing multimodal long-form summarization: A case study on financial reports. In *First Conference on Language Modeling*, 2024.

Leonardo Cella, Massimiliano Pontil, and Claudio Gentile. Best model identification: A rested bandit formulation. In *International Conference on Machine Learning*, pp. 1362–1372. PMLR, 2021.

Lingjiao Chen, Matei Zaharia, and James Zou. FrugalGPT: How to use large language models while reducing cost and improving performance. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Dujian Ding, Ankur Mallick, Chi Wang, Robert Sim, Subhabrata Mukherjee, Victor Rühle, Laks V. S. Lakshmanan, and Ahmed Hassan Awadallah. Hybrid LLM: Cost-efficient and quality-aware query routing. In *The Twelfth International Conference on Learning Representations*, 2024.

Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. Pal: program-aided language models. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.

Zeyu Han, Chao Gao, Jinyang Liu, Jeff Zhang, and Sai Qian Zhang. Parameter-efficient fine-tuning for large models: A comprehensive survey. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856.

Hoda Heidari, Michael J Kearns, and Aaron Roth. Tight policy regret bounds for improving and decaying bandits. In *IJCAI*, pp. 1562–1570, 2016.

Hugging Face. Hugging face. https://huggingface.co/, 2026. Accessed: 2026-04-18.

Nicole Immorlica, Jieming Mao, Aleksandrs Slivkins, and Zhiwei Steven Wu. Bayesian exploration with heterogeneous agents. WWW '19, pp. 751–761, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450366748.

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. Llm-blender: Ensembling large language models with pairwise ranking and generative fusion. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 14165–14178, 2023.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling Laws for Neural Language Models. 1 2020a.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020b.

Ilan Kremer, Yishay Mansour, and Motty Perry. Implementing the "wisdom of the crowd". *Journal of Political Economy*, 122(5):988–1012, 2014.

Toshiki Kuramoto and Jun Suzuki. Predicting fine-tuned performance on larger datasets before creating them. In *Proceedings of the 31st International Conference on Computational Linguistics: Industry Track*, pp. 204–212, 2025.

Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

Hongbo Li and Lingjie Duan. When congestion games meet mobile crowdsourcing: Selective information disclosure. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 5739–5746, 2023.

Hongbo Li and Lingjie Duan. Competitive multi-armed bandit games for resource sharing. *IEEE Transactions on Mobile Computing*, 2025.

Yang Li, Jiawei Jiang, Jinyang Gao, Yingxia Shao, Ce Zhang, and Bin Cui. Efficient automatic cash via rising bandits. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 4763–4771, 2020.

Yishay Mansour, Aleksandrs Slivkins, and Vasilis Syrgkanis. Bayesian incentive-compatible bandit exploration. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, pp. 565–582, 2015.

Alberto Maria Metelli, Francesco Trovo, Matteo Pirola, and Marcello Restelli. Stochastic rising bandits. In *International Conference on Machine Learning*, pp. 15421–15457. PMLR, 2022.

Quang H. Nguyen, Duy C. Hoang, Juliette Decugis, Saurav Manchanda, N. Chawla, and Khoa D. Doan. Metallm: A high-performant and cost-efficient dynamic framework for wrapping llms. *ArXiv*, abs/2407.10834, 2024. URL https://api.semanticscholar.org/CorpusID:271212682.

Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E. Gonzalez, M Waleed Kadous, and Ion Stoica. RouteLLM: Learning to route LLMs from preference data. In *The Thirteenth International Conference on Learning Representations*, 2025.

OpenAI. Openai api. https://platform.openai.com/, 2026. Accessed: 2026-04-18.

Manhin Poon, XiangXiang Dai, Xutong Liu, Fang Kong, John CS Lui, and Jinhang Zuo. Online multi-llm selection via contextual bandits under unstructured context evolution. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 24855–24863, 2026.

Yifan Sun, Jingyan Shen, Yibin Wang, Tianyu Chen, Zhendong Wang, Mingyuan Zhou, and Huan Zhang. Improving data efficiency for LLM reinforcement fine-tuning through difficulty-targeted online data selection and rollout replay. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=uwUkETPIJN.

Google DeepMind Team. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.

Qwen Team, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Together AI. Together ai platform. `https://docs.together.ai/`, 2026. Accessed: 2026-04-18.

Siwei Wang and Longbo Huang. Multi-armed bandits with compensation. *Advances in Neural Information Processing Systems*, 31, 2018.

Xu Wang, Yan Hu, Wenyu Du, Reynold Cheng, Benyou Wang, and Difan Zou. Towards understanding fine-tuning mechanisms of LLMs via circuit analysis. In *Forty-second International Conference on Machine Learning*, 2025.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, et al. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45. Association for Computational Linguistics, 2020.

Yu Xia, Fang Kong, Tong Yu, Liya Guo, Ryan A Rossi, Sungchul Kim, and Shuai Li. Which llm to play? convergence-aware online model selection with time-increasing bandits. In *Proceedings of the ACM Web Conference 2024*, pp. 4059–4070, 2024.

Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. A survey on multimodal large language models. *National Science Review*, 11(12), 2024. doi: 10.1093/nsr/nwae403.

Alex Young, Bei Chen, Chao Li, Chengen Huang, Ge Zhang, Guanwei Zhang, Guoyin Wang, Heng Li, Jiangcheng Zhu, Jianqun Chen, et al. Yi: Open foundation models by 01. ai. *arXiv preprint arXiv:2403.04652*, 2024.

Zheng Yuan, Hongyi Yuan, Chengpeng Li, Guanting Dong, Keming Lu, Chuanqi Tan, Chang Zhou, and Jingren Zhou. Scaling relationship on learning mathematical reasoning with large language models. *arXiv preprint arXiv:2308.01825*, 2023.

Biao Zhang, Zhongtao Liu, Colin Cherry, and Orhan Firat. When scaling meets llm finetuning: The effect of data, model and finetuning method. In *International Conference on Learning Representations*, volume 2024, pp. 44694–44713, 2024.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*, 2020.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.