PULSE: PROVENANCE-AWARE USER-PROFILE LEARNING AND SELECTIVE EVALUATION FOR LONG-TERM LLM PERSONALIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-term LLM personalization aims to adapt an assistant's responses to user preferences as they evolve across interactions. Existing approaches typically represent users through latent embeddings, natural-language profiles, or learned summaries to guide future response generation based on interaction history. However, profile updating and utilization often remain implicit, leading to two challenges: situational or uninformative feedback may be incorporated into persistent preferences, while valid but context-irrelevant profile information may distort the evaluation of responses to the current request. Long-term personalization therefore requires determining when recent interactions provide reusable preference evidence that justifies profile revision and when stored information should influence response selection. To address these challenges, we propose PULSE, a provenance-aware framework that explicitly models both profile updating from interaction evidence and selective profile utilization in personalized response evaluation. PULSE selects interaction events containing reusable preference evidence, determines the appropriate update state (no update, local preference refinement, or long-term preference revision), and generates evidence-grounded profile statements. It then converts updated profiles into request-specific criteria and evaluates candidate responses through a criterion-guided personalized residual reward. This residual reward selectively corrects a profile-free anchor reward when the profile provides applicable guidance, preserving general response quality while preventing irrelevant preferences from distorting response selection. Experiments on HorizonBench with both static- and evolving-preference instances show that PULSE achieves effective personalized ranking and preference-change discrimination while maintaining evidence-grounded profiles and limiting the influence of unsupported or context-irrelevant profile information. Code is available at `https://anonymous.4open.science/r/pulse-FBE3`

1 INTRODUCTION

Large language models (LLMs) are increasingly expected to serve as persistent assistants that adapt to users across repeated interactions (Zhong et al., 2024; Li et al., 2025; Uddin et al., 2026). Existing personalization methods represent users through latent embeddings (Poddar et al., 2024), natural-language personas (Ryan et al., 2025), learned summaries or evolving profiles (Nam et al., 2026; Zhao et al., 2025). These representations allow past interactions to inform future responses, but reliable long-term personalization requires more than accumulating user information. A user's feedback may reveal a stable preference, express a task-specific need, or provide no reusable preference information at all. Moreover, even a correctly stored preference should influence a response only when it is relevant to the current request. The central challenge is therefore not simply what an assistant should remember, but when the user profile should be revised and when its contents should affect response selection.

As illustrated in Figure 1, conventional profile-based personalization can exhibit two distinct failure modes. On the profile-update side, a user asks for a short answer because they are currently in a hurry, but the system overgeneralizes this situational request into a persistent preference for concise responses. On the profile-utilization side, the profile correctly records that the user prefers

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.

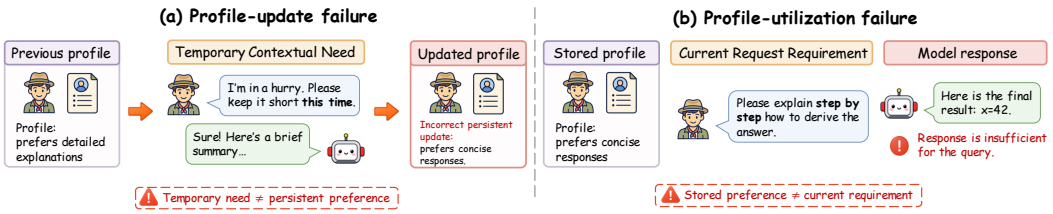


Figure 1: Motivating examples of two failures in profile-based LLM personalization. (a) **Failure on the profile-update side**: a temporary request for brevity is overgeneralized into a persistent profile update, overwriting the user's long-term preference for detailed explanations. (b) **Failure on the profile-utilization side**: a stored preference for concise responses is applied to a request that explicitly requires a step-by-step explanation, resulting in an insufficient response.

concise summaries, yet the system applies this preference to a request that requires a complete step-by-step explanation and consequently favors an insufficient response. These failures expose two decisions often hidden inside a single profile-conditioned pipeline: whether recent interactions contain reusable preference evidence that justifies changing the profile, and whether a stored profile statement applies to the current request. This distinction is consistent with recent findings that LLMs tend to over-select user attributes as relevant and that valid user information does not always improve personalized responses (Zhang et al., 2026a). Reliable personalization therefore requires explicit control over both how user profiles are updated and how they are subsequently used.

To address these two problems, we propose PULSE, a provenance-aware user-profile learning and selective evaluation framework built around two coupled components for profile updating and profile utilization. Our first component, **Factorized Tri-State Profile Update**, factorizes profile revision into reusable-evidence selection, update-state attribution, and evidence-grounded profile generation. Given a previous profile and recent interactions, PULSE selects interaction evidence containing reusable preference information, predicts one of three update states (no update, local preference refinement, or long-term preference revision), and generates the updated profile conditioned on the selected evidence and state. Newly introduced or materially revised profile statements must cite their supporting interaction evidence. By explicitly separating reusable-evidence selection from update-state attribution, PULSE prevents a situational request from being implicitly treated as a persistent preference while allowing genuine preference changes to revise outdated profile statements.

Our second component, **Criterion-Guided Personalized Residual Reward**, introduces an anchor-residual decomposition that separates profile-independent response quality from request-specific personalized alignment. Given the updated evidence-state profile and the current request, PULSE converts relevant profile statements into explicit evaluation criteria. A profile-free anchor reward captures the general quality of each candidate response, while a criterion-guided residual measures the candidate's additional alignment with the applicable personalized criteria. The final reward combines these two signals, so profile information affects response ranking only when it yields an applicable personalized criterion. Coupling evidence-attributed profile updating with selective residual reward modeling yields a traceable path from interaction evidence, through scoped profile updates and request-specific criteria, to response ranking.

Our main contributions are summarized as follows:

- We formulate long-term LLM personalization as two coupled problems: deciding when interaction evidence warrants revising the user profile and which profile statements should influence the current response.
- We propose PULSE, which couples **Factorized Tri-State Profile Update** for evidence-grounded and scope-aware profile revision with **Criterion-Guided Personalized Residual Reward** for selective personalized response evaluation.
- Extensive experiments on HorizonBench under both static and evolving preference settings demonstrate that PULSE provides effective personalized ranking and preference-change discrimination while producing evidence-grounded profile updates and applying profile information selectively.

2 PROBLEM FORMULATION

We study long-term LLM personalization, where an assistant must update its understanding of a user from historical interactions and use the resulting profile to evaluate responses to future requests. We consider one profile-update step for an arbitrary user and omit user and time indices for clarity. Let $\mathcal{P}^- = \{p_i\}_{i=1}^M$ denote the profile before the update, and let $\mathcal{H} = \{h_i\}_{i=1}^L$ denote a chronologically ordered window of recent interactions. Each interaction $h_i = (x_i, y_i, f_i)$ consists of a user request x_i, an assistant response y_i, and the subsequent user feedback f_i. A profile updater produces $(\mathcal{E}, z, \mathcal{P}^+) = \mathcal{U}_\phi(\mathcal{P}^-, \mathcal{H})$, where $\mathcal{E} \subseteq \mathcal{H}$ is the selected reusable preference evidence, $\mathcal{P}^+$ is the updated profile, and $z \in \{\mathrm{N}, \mathrm{L}, \mathrm{R}\}$ denotes the update state. Here, N represents *no update*, L represents *local preference refinement*, and R represents *long-term preference revision*. The three states indicate, respectively, that the recent interactions contain no reusable preference information, support only a context-bound refinement, or justify revising a persistent preference. For $z = \mathrm{N}$, the previous profile should be preserved. Otherwise, every newly introduced or materially revised statement $p \in \mathcal{P}^+$ is associated with a nonempty provenance set $\pi(p) \subseteq \mathcal{E}$, identifying the interaction evidence that supports the change. We denote the resulting evidence-state profile by $\mathcal{Z} = (\mathcal{P}^+, \mathcal{E}, z)$. Given a new request x and a candidate response set $\mathcal{Y} = \{y_1, \ldots, y_K\}$, a personalized reward model assigns each candidate a scalar score $r_\theta(y \mid x, \mathcal{Z})$, and selects the response with the highest score. For a preferred response y^+ and a rejected response y^-, their pairwise preference probability is modeled as

$$P_\theta(y^+ \succ y^- \mid x, \mathcal{Z}) = \sigma\left(r_\theta(y^+ \mid x, \mathcal{Z}) - r_\theta(y^- \mid x, \mathcal{Z})\right),$$

where $\sigma(\cdot)$ is the sigmoid function. Accordingly, each learning instance can be represented as $(\mathcal{P}^-, \mathcal{H}, x, y^+, y^-) \in \mathcal{D}$.

Our goal is to learn a personalized system that jointly performs profile updating and response evaluation. Specifically, we aim to: (1) revise the user profile only when reusable interaction evidence supports an appropriate update scope, while preserving explicit provenance for every substantive change; and (2) correctly rank candidate responses using the updated profile, while preventing unsupported or context-irrelevant profile information from influencing the current decision. Ultimately, the system should maintain an evidence-grounded representation of evolving user preferences and use it selectively for reliable long-term personalization.

3 METHOD

In this section, we introduce PULSE, a provenance-aware user-profile learning and selective evaluation framework for long-term LLM personalization. As shown in Figure 2, PULSE consists of two coupled components: (1) **Factorized Tri-State Profile Update**, which transforms the previous profile and recent interactions into an evidence-state profile by sequentially selecting reusable evidence, determining the scope of the update, and generating an evidence-grounded profile; and (2) **Criterion-Guided Personalized Residual Reward**, which derives request-specific criteria from the updated profile and uses them to selectively refine a profile-free anchor reward for personalized response ranking.

3.1 FACTORIZED TRI-STATE PROFILE UPDATE

A reliable profile updater must determine not only what information should be retained, but also whether the observed evidence reflects a situational requirement or a persistent preference change. Directly generating a new profile from recent interactions entangles these decisions, making it easy to convert temporary requests into long-term preferences. PULSE instead represents profile updating as an ordered evidence–state–profile process.

Factorized update trace. Given the previous profile $\mathcal{P}^-$ and a chronological window of recent interactions $\mathcal{H}$, the updater produces three outputs: a reusable evidence subset $\mathcal{E} \subseteq \mathcal{H}$, an update state z, and the complete updated profile $\mathcal{P}^+$. Their generation is factorized as

$$\begin{aligned} p_\phi(\mathcal{E}, z, \mathcal{P}^+ \mid \mathcal{P}^-, \mathcal{H}) = p_\phi(\mathcal{E} \mid \mathcal{P}^-, \mathcal{H}) \\ \cdot p_\phi(z \mid \mathcal{P}^-, \mathcal{H}, \mathcal{E}) \\ \cdot p_\phi(\mathcal{P}^+ \mid \mathcal{P}^-, \mathcal{H}, \mathcal{E}, z). \end{aligned} \tag{1}$$

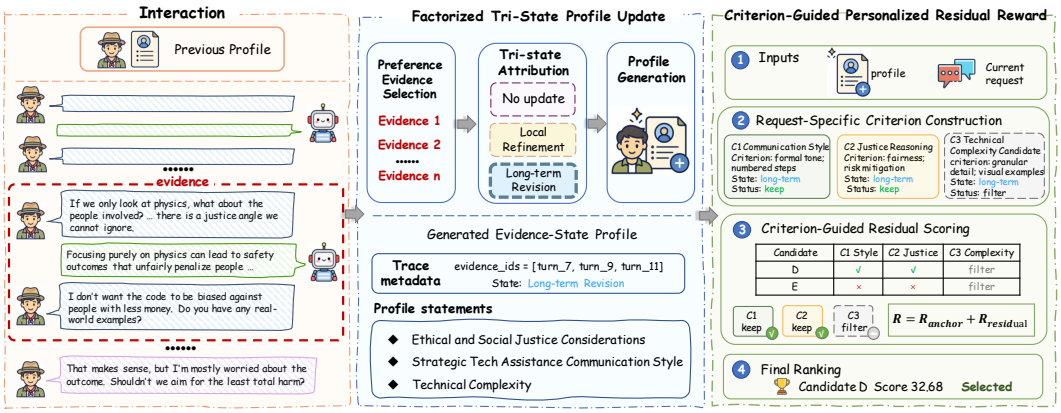


Figure 2: Overview of PULSE. Given a history profile and recent interactions, the Factorized Tri-State Profile Update module selects reusable preference evidence, determines its update scope, and generates an evidence-grounded profile with traceable update metadata. For a current request, the Criterion-Guided Personalized Residual Reward module derives request-specific criteria from the updated profile and uses them to produce a personalized residual. This residual selectively refines a profile-free anchor reward to rank candidate responses.

The updater first selects the interaction IDs that contain reusable preference information and then assigns one of three states, $z \in \{\mathrm{N}, \mathrm{L}, \mathrm{R}\}$. Here, N denotes *no update*, L denotes *local preference refinement*, and R denotes *long-term preference revision*. The three states distinguish irrelevant or non-reusable evidence, context-bound preferences, and evidence that justifies modifying a persistent preference, respectively.

Evidence-grounded profile generation. Conditioned on the selected evidence and update state, the updater generates the complete next profile rather than an isolated edit instruction. For $z = \mathrm{N}$, it returns the previous profile without introducing new preference information. For $z = \mathrm{L}$, it may record a context-bound preference without replacing an established long-term preference. For $z = \mathrm{R}$, it may add, rewrite, merge, or remove profile statements to reflect a persistent preference change. Every newly introduced or materially revised statement must cite at least one interaction ID from $\mathcal{E}$, while retained statements preserve their existing provenance. The resulting evidence-state profile is denoted by $\mathcal{Z} = (\mathcal{P}^+, \mathcal{E}, z)$.

Learning the updater. To train the updater without requiring manually annotated update states, we sample multiple structured update traces for each update context and additionally include the identity trace that preserves the previous profile. Each candidate trace is evaluated using training-split response-preference labels and automatic evidence-attribution checks. Its utility measures improvements in current and future personalized response ranking, discrimination between current and stale preferences, evidence-attribution consistency, and unnecessary profile growth. Candidates with higher utility are paired against meaningfully weaker candidates, and the updater is trained with DPO on the complete serialized traces. Because evidence selection, update state, and profile generation occur in the same ordered output, their supervision comes from the utility of the complete trace rather than comparison against reference update-state labels. Appendix C.1 defines the utility components and preference-pair construction.

3.2 Criterion-Guided Personalized Residual Reward

Even an accurately updated profile may contain statements that should not affect every request. Directly conditioning a reward model on the entire profile can therefore over-apply valid but contextually irrelevant preferences. PULSE addresses this problem by separating profile-independent response quality from personalized preference alignment: the former is captured by an anchor reward, while the latter is introduced only through criteria derived for the current request.

Request-specific criterion construction. Given the evidence-state profile $\mathcal{Z}$ and the current request x, PULSE constructs a compact criterion set $\mathcal{C}_x = \{c_1, \ldots, c_K\}$. Profile statements are first filtered according to their provenance and update state, excluding statements that are neither inherited from the previous profile nor supported by the selected evidence. PULSE then identifies the statements applicable to the current request and converts them into explicit, observable evaluation criteria. Each criterion remains grounded in its source profile statement, while irrelevant profile information is excluded from subsequent response evaluation.

Profile-free anchor and personalized residual. For each candidate response y_i, PULSE computes a profile-free anchor score r_i^{anc} that captures its general quality with respect to the current request. In parallel, the request-specific criterion set is used to estimate a personalized residual r_i^{res}, which reflects how well the candidate aligns with the applicable user preferences. The final reward is defined as

$$r_i = r_i^{\text{anc}} + r_i^{\text{res}}. \tag{2}$$

When the profile provides no applicable criterion, the residual is omitted and response ranking is determined by the anchor reward. This formulation prevents personalization from replacing general response quality and instead treats it as a selective correction.

Reward learning. The anchor reward is first learned to capture profile-independent response quality and is then kept fixed during personalized reward learning. The criterion construction module is supervised by a training-only teacher that observes the attributed profile, current request, and chosen–rejected response contrasts. The teacher's structured criterion decisions serve as targets for supervised fine-tuning of a separate criterion adapter, whose inputs contain only the attributed profile, update state, and current request. The residual reward is then trained using criteria generated by this adapter through the final response-ranking objective. This separation preserves a stable quality reference and confines user-specific influence to the criterion-guided residual.

Specifically, given a candidate set $\mathcal{Y}$ containing the preferred response y^+, we optimize the criterion-guided residual through the final ranking objective:

$$\mathcal{L}_{\text{rank}} = -\mathbb{E}_{(\mathcal{Y}, y^+)} \left[\log \frac{\exp\big(r_{y^+}\big)}{\sum_{y \in \mathcal{Y}} \exp\big(r_y\big)} \right]. \tag{3}$$

Because the anchor reward remains fixed at this stage, the objective learns only the personalized correction needed to improve candidate ranking under the applicable criteria. Criterion supervision therefore requires no human-authored criterion annotations, and response contrasts are used only to construct training targets. Appendix H details the teacher–student training procedure and prompts. At inference time, candidate responses are ranked by their final anchor-plus-residual scores, completing a traceable path from interaction evidence and profile updates to personalized response selection.

4 EXPERIMENTS

We conduct experiments to systematically evaluate the effectiveness of PULSE for long-term LLM personalization. Beyond overall response-ranking performance, we investigate the contributions of its profile-update and profile-utilization components, examine whether it produces accurate and evidence-grounded profile updates, and analyze whether its criterion-guided residual reward introduces useful personalized corrections while preserving general response quality.

Research Questions. Our experiments are designed to answer the following research questions:

- **RQ1 (Overall Effectiveness).** How effectively does PULSE rank responses aligned with users' current preferences across seen and unseen users, static and evolving preferences, and different backbone models, compared with representative baselines?
- **RQ2 (Ablation Study).** How do PULSE's profile-update and profile-utilization components contribute, individually and jointly, to its personalized ranking performance?

- **RQ3 (Profile Update Quality).** Can PULSE produce accurate and evidence-grounded profile updates by selecting reusable preference evidence, assigning appropriate update states, and avoiding unsupported or stale profile statements?
- **RQ4 (Selective Residual Reward).** Does the criterion-guided personalized residual improve response ranking when profile information is applicable while avoiding unnecessary corrections when stored preferences are irrelevant to the current request?

4.1 Experimental Setup

Dataset. We evaluate PULSE on HorizonBench (Li et al., 2026), a long-horizon personalization benchmark constructed from multi-turn user–assistant interactions. The benchmark contains 4,245 evaluation instances from 346 users across 30 preference domains. Each instance provides a current user request and up to five candidate responses, with one response aligned with the user's current preference. Instances are categorized as *static*, where the relevant preference remains unchanged, or *evolving*, where the current preference differs from an earlier one. A subset of the evolving instances additionally contains an annotated old-preference distractor, allowing us to evaluate whether a model follows the current rather than an obsolete preference.

We consider two complementary evaluation settings. In the *seen-user* setting, each user's interactions are divided chronologically: earlier instances are used for training, while later instances are reserved for validation and testing. This setting evaluates whether a model can continually adapt to the evolving preferences of users observed during training. The resulting split contains 3,376 training, 373 validation, and 496 test instances. In the *unseen-user* setting, complete user histories are assigned to mutually exclusive splits, yielding 3,516 training, 341 validation, and 388 test instances from 286, 30, and 30 non-overlapping users, respectively. This setting evaluates whether the learned personalization mechanism generalizes to users absent from model training. The unseen-user setting is not history-free: the model may observe a test user's preceding interactions at inference time, but no instances or preference labels from that user are used to optimize model parameters.

For each instance, PULSE receives the previous user profile, at most eight chronologically preceding interactions containing explicit user feedback, the current request, and the candidate responses. The earliest available preference snapshot initializes the previous profile, which is subsequently updated autoregressively using PULSE's own predictions without reference-profile teacher forcing. Current-preference labels, old-preference annotations, and future interactions are used only to construct training objectives or evaluation metrics and are never exposed during test-time ranking.

Baselines. We compare PULSE with three groups of representative baselines.

- **Context-based methods.** BASE LLM directly ranks the candidate responses without user-specific adaptation. RAG retrieves the most relevant previous interactions from the same user and includes them as additional context for candidate ranking.
- **General alignment methods.** SFT fine-tunes the backbone model on preferred responses, while DPO learns from preferred–rejected response pairs through direct preference optimization (Rafailov et al., 2023). These methods learn general response preferences but do not explicitly maintain an evolving user profile.
- **Personalized methods.** VPL learns user-specific latent representations for personalized reward modeling (Poddar et al., 2024). PAL performs sample-efficient personalized reward adaptation from limited user feedback (Chen et al., 2025). PLUS learns textual user summaries from historical interactions and uses them to condition a personalized reward model (Nam et al., 2026).

Evaluation Metrics. We report Top-1 Accuracy (T1A) and Mean Reciprocal Rank (MRR) on the overall, static, and evolving subsets. On evolving instances with an old-preference distractor, Current-over-Old Accuracy (COA) measures how often the current-preference response outranks the obsolete-preference response. Formal definitions are provided in Appendix D.

4.2 Overall Effectiveness (RQ1)

PULSE consistently improves overall response ranking. As shown in Table 1, PULSE achieves the highest overall T1A and MRR across backbones under seen and unseen settings. Although

Table 1: Performance comparison under the seen and unseen settings with Qwen3-4B-Instruct-2507 and Llama-3.2-3B-Instruct as backbone models. Best and second-best results are shown in **bold** and underlined, respectively.

Setting	Method	T1A ↑			MRR ↑			COA ↑
		All	Evolved	Static	All	Evolved	Static	Evolved
Qwen3-4B-Instruct-2507								
Seen	Base	0.7358	0.7279	0.7455	0.8413	0.8365	0.8471	0.8555
	SFT	0.7983	0.7867	0.8125	0.8819	0.8751	0.8901	0.8722
	RAG (BGE-M3)	0.8870	0.8897	0.8839	0.9316	0.9348	0.9277	**0.9666**
	DPO	0.8407	0.8492	0.8303	0.9131	0.9175	0.9078	0.9277
	VPL	0.8891	0.8823	0.8973	0.9400	0.9348	0.9464	0.9333
	PAL	0.7883	0.7720	0.8080	0.8778	0.8675	0.8902	0.9277
	PLUS	0.8874	0.8575	**0.9237**	0.9370	0.9227	**0.9544**	0.9333
	PULSE	**0.9052**	**0.9044**	0.9063	**0.9466**	**0.9453**	0.9479	0.9611
Unseen	Base	0.6984	0.7169	0.6761	0.8152	0.8238	0.8048	0.8828
	SFT	0.7654	0.7830	0.7443	0.8620	0.8702	0.8521	0.9062
	RAG (BGE-M3)	0.8324	0.8349	0.8295	0.8949	0.8952	0.8946	0.9375
	DPO	0.8582	0.8584	0.8579	0.9217	0.9215	0.9220	0.9218
	VPL	0.8737	0.8773	0.8693	0.9298	0.9325	0.9266	0.9375
	PAL	0.7540	0.7121	0.8017	0.8538	0.8304	0.8804	0.8648
	PLUS	0.8505	0.8349	0.8693	0.9144	0.9087	0.9212	**0.9453**
	PULSE	**0.8995**	**0.8915**	**0.9091**	**0.9436**	**0.9365**	**0.9522**	0.9219
Llama-3.2-3B-Instruct								
Seen	Base	0.7580	0.75	0.7678	0.8250	0.8397	0.8073	0.85
	SFT	0.7883	0.7647	0.8169	0.8759	0.8619	0.8930	0.8555
	RAG (BGE-M3)	0.8850	0.8786	0.8928	0.9309	0.9275	0.9350	0.9166
	DPO	0.8991	0.8860	0.9151	0.9439	0.9348	0.9549	0.9555
	VPL	0.8689	0.8639	0.875	0.9253	0.9242	0.9267	0.9222
	PAL	0.8245	0.8198	0.8303	0.8958	0.8906	0.9021	0.9277
	PLUS	0.8851	0.8625	0.9125	0.9349	0.9205	0.9523	0.9333
	PULSE	**0.9268**	**0.9013**	**0.9369**	**0.9554**	**0.9421**	**0.9638**	**0.9568**
Unseen	Base	0.7061	0.7358	0.6704	0.8250	0.8397	0.8073	0.8984
	SFT	0.75	0.7641	0.7329	0.8558	0.8596	0.8512	0.8828
	RAG (BGE-M3)	0.8402	0.8490	0.8295	0.9054	0.9091	0.9009	0.9218
	DPO	0.8737	0.8584	0.8920	0.9288	0.9199	0.9395	0.9218
	VPL	0.8659	0.8396	0.8977	0.9265	0.9128	0.9428	0.9140
	PAL	0.8024	0.7727	0.8362	0.8848	0.8656	0.9066	0.8513
	PLUS	0.8505	0.8349	0.8693	0.9164	0.9071	0.9275	0.9296
	PULSE	**0.9083**	**0.8813**	**0.9321**	**0.9436**	**0.9352**	**0.9569**	**0.9534**

the strongest baseline varies across configurations, PULSE consistently ranks first on both metrics. These results indicate that its ranking advantage holds across the backbones and user settings.

The gains extend to unseen users. On unseen users, PULSE achieves T1A scores of 0.8995 with Qwen and 0.9083 with Llama, exceeding the strongest corresponding baselines by 2.58 and 3.46 percentage points. These margins are larger than those in the seen setting, suggesting that the benefits extend beyond users encountered during training. The strongest baseline also changes across backbones: VPL leads overall T1A on Qwen, whereas DPO leads on Llama. PULSE outperforms both, supporting the consistency of its ranking advantage across model families.

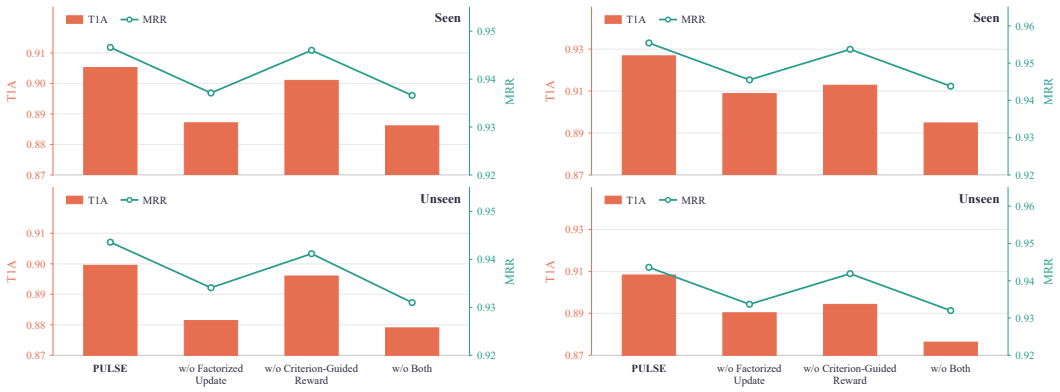
Historical preference information alone does not ensure accurate personalization. RAG substantially improves over Base, demonstrating the value of user interaction history. SFT and DPO improve ranking, but their training objectives do not explicitly distinguish enduring preferences from temporary or superseded ones. VPL and PAL capture user heterogeneity through latent representations, while PLUS constructs textual user representations; these representations do not establish whether a historical preference should govern the current comparison. This provides a plausible explanation for their remaining errors: retrieving or summarizing a preference does not necessar-

ily resolve its current applicability. PULSE's gains are consistent with addressing this distinction through scoped profile updates and comparison-specific criterion verification.

Adaptation gains coexist with strong static performance. PULSE achieves the highest evolved-subset T1A and MRR in all four settings, while retaining the strongest static-subset performance in three. However, its advantage varies across metrics: PLUS leads static ranking on Qwen-seen, and RAG and PLUS achieve higher COA on Qwen-seen and Qwen-unseen, respectively. PULSE leads COA in both Llama settings. These results support a consistent advantage in complete candidate ranking, with improvements in current-over-old discrimination depending on the backbone.

4.3 Ablation Study (RQ2)

To examine the contributions of PULSE's two components, we compare the full model with three variants using the same backbone. *w/o Factorized Update* directly generates the updated profile from the previous profile and recent interactions, replacing the factorized evidence-selection and tri-state update process while retaining the criterion-guided reward. *w/o Criterion-Guided Reward* retains the proposed profile updater but trains a reward model directly on the entire updated profile, without constructing request-specific criteria or separating the reward into an anchor and a personalized residual. *w/o Both* combines direct profile updating with this full-profile reward model. These variants assess the contribution of each component and their combined effect on response ranking.



(a) Qwen3-4B-Instruct-2507 (b) Llama-3.2-3B-Instruct

Figure 3: Ablation results under seen and unseen settings.

As shown in Figure 3, replacing the factorized updater with direct profile updating reduces performance most clearly on evolving instances, consistent with the loss of explicit evidence and update-state decisions. Replacing the criterion-guided reward with direct full-profile scoring also weakens ranking, suggesting that an updated profile alone does not ensure its appropriate use for the current request. The full model performs best, whereas replacing both components leads to the largest degradation. These results support the complementary roles of structured profile updating and selective personalized reward modeling.

4.4 Profile Update Quality (RQ3)

To evaluate profile-update quality, we use 175 test instances from 18 unseen users with Qwen3-4B-Instruct-2507. Two independent GPT-5.5 annotation runs, followed by separate adjudication, yield reference evidence, update states, and profile statements. We compare PULSE with No Update, which retains the prior profile; Direct Update, which rewrites it from recent interactions; and Prompted Tri-State, which predicts an update state through prompting before generating the profile. Annotation and semantic-matching details are provided in Appendix G.

We evaluate the complete update process from four complementary aspects. ES-F1 measures whether reusable preference evidence is correctly selected, while US-F1 evaluates the attribution of update states. CS-F1 focuses on the profile changes introduced by each method and is computed on instances requiring an update, avoiding domination by unchanged historical statements.

Table 2: Evaluation of profile update quality.

Method	ES-F1↑	US-F1↑	CS-F1↑	ESR↑
NO UPDATE	–	–	0.0	–
DIRECT UPDATE	–	–	15.4	0.0
PROMPTED TRI-STATE	**75.7**	40.8	50.2	72.4
PULSE	72.8	**54.3**	**54.1**	**82.2**

ESR further evaluates whether newly introduced or materially revised statements are supported by their cited interactions. As shown in Table 2, PULSE achieves the best update-state attribution, changed-statement accuracy, and evidence support rate. Compared with NO UPDATE and DIRECT UPDATE, it more accurately captures the profile changes warranted by recent interactions. Although PROMPTED TRI-STATE obtains higher ES-F1 through broader evidence selection, PULSE yields more reliable update-state decisions and more consistently grounded profile revisions.

4.5 SELECTIVE RESIDUAL REWARD (RQ4)

Using the same Qwen3-4B-Instruct-2507 unseen-user subset as RQ3, we examine whether PULSE applies profile information selectively and whether its residual improves response ranking. Two independent GPT-5.5 judging runs determine whether each request warrants personalization and whether every generated criterion is grounded in its source profile statement, applicable to the current request, and nonredundant with information explicitly stated in the request. Disagreements are resolved through a separate adjudication pass. A criterion decision is considered correct if PULSE emits a nonempty set containing only valid criteria when personalization is warranted, or abstains when it is not. We then compare the final reward with its profile-free anchor and count beneficial and harmful Top-1 ranking flips. Further details are provided in Appendix G.

Table 3: Criterion selectivity and residual effects on response ranking. Decision Acc. measures correct criterion selection or abstention, while Helped and Hurt report beneficial and harmful Top-1 ranking flips relative to the profile-free anchor.

Request type	Decision Acc.↑	Helped	Hurt
Personalization warranted	80.0	4	0
No personalization warranted	72.0	0	1

Table 3 shows that criterion selectivity generally translates into appropriate reward corrections. When personalization is warranted, PULSE achieves 80.0% decision accuracy and corrects four anchor errors without introducing any harmful Top-1 flip. When personalization is not warranted, it correctly abstains in 72.0% of the instances; most anchor rankings therefore remain unchanged, although one unnecessary residual intervention introduces a ranking error. These results show that the residual is most useful when profile information is applicable, while also identifying criterion abstention as the main remaining source of harmful corrections.

5 CONCLUSION

We presented PULSE to address two coupled decisions in long-term LLM personalization: when interaction evidence warrants revising a user profile and when stored preferences should influence response evaluation. PULSE combines a factorized tri-state updater that grounds profile revisions in reusable evidence with a criterion-guided residual reward that selectively refines a profile-free anchor for the current request. Experiments on HorizonBench show that PULSE improves personalized response ranking across seen and unseen users under both static and evolving preferences. Ablation and component analyses further show that factorized updating produces more accurate, evidence-grounded profiles, while the criterion-guided residual uses those profiles selectively to improve response ranking. Together, these findings demonstrate the value of coordinating when a user profile should change with when its contents should influence a response.

6 AI USE STATEMENT

In this work, we used generative AI tools to assist with code development and manuscript editing for readability. We also used GPT-5.5 to produce independent annotations for the profile-update and criterion-selectivity evaluations, and GPT-5.6 Sol to adjudicate disagreements between annotation runs, as described in Appendix G. These model-based annotations were generated using predefined rubrics, with method identities and ranking outcomes hidden where applicable. All LLM-generated code was verified and tested by the authors, and all AI-edited text was checked for factual accuracy and originality. The authors take full responsibility for the experimental protocols, reported results, and final content of the paper.

REFERENCES

Avinandan Bose, Zhihan Xiong, Yuejie Chi, Simon Shaolei Du, Lin Xiao, and Maryam Fazel. Lore: Personalizing llms via low-rank reward modeling, 2025. URL https://arxiv.org/abs/2504.14439.

Daiwei Chen, Yi Chen, Aniket Rege, Zhi Wang, and Ramya Vinayak. Pal: Sample-efficient personalized reward modeling for pluralistic alignment. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 15755–15799, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/2858f8c8683aaa8c12d487354cf328dc-Paper-Conference.pdf.

Wonjun Choi, Yerim Kim, Yukyung Lee, and Susik Yoon. Pgmem: Tightly coupled persona-memory graph for lifelong personalized agents. *arXiv preprint arXiv:2608.01708*, 2026.

Youngbin Choi, Seunghyuk Cho, Minjong Lee, MoonJeong Park, Yesong Ko, Jungseul Ok, and Dongwoo Kim. CoPL: Collaborative preference learning for personalizing LLMs. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 12875–12893, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.650. URL https://aclanthology.org/2025.emnlp-main.650/.

Linfeng Du, Ye Yuan, Zichen Zhao, Fuyuan Lyu, Emiliano Penaloza, Xiuying Chen, Zipeng Sun, Jikun Kang, Laurent Charlin, Xue Liu, and Haolun Wu. Optimizing user profiles via contextual bandits for retrieval-augmented LLM personalization. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 31803–31817, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.1467. URL https://aclanthology.org/2026.acl-long.1467/.

Zhaoxin Feng, Jianfei Ma, and Emmanuele Chersoni. Know it, act on it: Investigating memory utilization in llm personalization. *arXiv preprint arXiv:2607.29433*, 2026.

Bowen Jiang, Zhuoqun Hao, Young-Min Cho, Bryan Li, Yuan Yuan, Sihao Chen, Lyle Ungar, Camillo J Taylor, and Dan Roth. Know me, respond to me: Benchmarking llms for dynamic user profiling and personalized responses at scale. *arXiv preprint arXiv:2504.14225*, 2025a.

Bowen Jiang, Yuan Yuan, Maohao Shen, Zhuoqun Hao, Zhangchen Xu, Zichen Chen, Ziyi Liu, Anvesh Rao Vijjini, Jiashu He, Hanchao Yu, et al. Personamem-v2: Towards personalized intelligence via learning implicit user personas and agentic memory. *arXiv preprint arXiv:2512.06688*, 2025b.

Ishita Kumar, Snigdha Viswanathan, Sushrita Yerra, Alireza Salemi, Ryan A Rossi, Franck Dernoncourt, Hanieh Deilamsalehy, Xiang Chen, Ruiyi Zhang, Shubham Agarwal, et al. Longlamp: A benchmark for personalized long-form text generation. *arXiv preprint arXiv:2407.11016*, 2024.

Hao Li, Chenghao Yang, An Zhang, Yang Deng, Xiang Wang, and Tat-Seng Chua. Hello again! LLM-powered personalized agent for long-term dialogue. In Luis Chiruzzo, Alan Ritter, and

Lu Wang (eds.), *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 5259–5276, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-189-6. doi: 10.18653/v1/2025.naacl-long.272. URL https://aclanthology.org/2025.naacl-long.272/.

Shuyue Stella Li, Bhargavi Paranjape, Kerem Oktar, Zhongyao Ma, Gelin Zhou, Lin Guan, Na Zhang, Sem Park, Lin Chen, Diyi Yang, Yulia Tsvetkov, and Asli Celikyilmaz. Horizonbench: Long-horizon personalization with evolving preferences, 2026. URL https://arxiv.org/abs/2604.17283.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

HyunJi Nam, Yanming Wan, Mickel Liu, Peter Ahnn, Jianxun Lian, and Natasha Jaques. Learning to summarize user information for personalized reinforcement learning from human feedback. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 122761–122784, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/c7af751b5e0a407c62ac023e3cb381f9-Paper-Conference.pdf.

Sriyash Poddar, Yanming Wan, Hamish Ivison, Abhishek Gupta, and Natasha Jaques. Personalizing reinforcement learning from human feedback with variational preference learning. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 52516–52544. Curran Associates, Inc., 2024. doi: 10.52202/079017-1664. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/5e1c255653eb98cef13f45b2d337c882-Paper-Conference.pdf.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

Michael J Ryan, Omar Shaikh, Aditri Bhagirath, Daniel Frees, William Held, and Diyi Yang. Synthesizeme! inducing persona-guided prompts for personalized reward models in llms. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8045–8078, 2025.

Alireza Salemi, Sheshera Mysore, Michael Bendersky, and Hamed Zamani. LaMP: When large language models meet personalization. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7370–7392, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.399. URL https://aclanthology.org/2024.acl-long.399/.

Kwangwook Seo and Dongha Lee. P-check: Advancing personalized reward model via learning to generate dynamic checklist. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 43447–43471, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.2011. URL https://aclanthology.org/2026.acl-long.2011/.

Jiajia Song, Bobo Li, Haiwen Yi, Zibo Ji, Meishan Zhang, Hao Fei, Min Zhang, Mong-Li Lee, and Wynne Hsu. From profiling to synthesis: Benchmarking implicit behavioral alignment in personalized llm agents. *arXiv preprint arXiv:2608.02171*, 2026.

Md Nayem Uddin, Kumar Shubham, Eduardo Blanco, Chitta Baral, and Gengyu Wang. From recall to forgetting: Benchmarking long-term memory for personalized agents. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 26814–26841, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.1337. URL https://aclanthology.org/2026.findings-acl.1337/.

Jiaxing Wu, Lin Ning, Luyang Liu, Harrison Lee, Neo Wu, Chao Wang, Sushant Prakash, Shawn O'Banion, Bradley Green, and Jun Xie. Rlpf: Reinforcement learning from prediction feedback for user summarization with llms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 25488–25496, 2025.

Jianfei Xiao, Xiang Yu, Chengbing Wang, Wuqiang Zheng, Xinyu Lin, Kaining Liu, Hongxun Ding, Yang Zhang, Wenjie Wang, Fuli Feng, and Xiangnan He. Alpsbench: An llm personalization benchmark for real-dialogue memorization and preference alignment. In *Proceedings of the 49th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '26, pp. 3489–3497, New York, NY, USA, 2026. Association for Computing Machinery. ISBN 9798400725999. doi: 10.1145/3805712.3808634. URL https://doi.org/10.1145/3805712.3808634.

Lechen Zhang, Jiarui Liu, and Tal August. Re-centering humans in llm personalization. *arXiv preprint arXiv:2606.06614*, 2026a.

Pinyi Zhang, Ting-En Lin, Yuchuan Wu, Jingyang Chen, Zongqi Wang, Hua Yang, Bing Zhao, Fei Huang, Yongbin Li, and Kai Zhang. P-genrm: Personalized generative reward model with test-time user-based scaling. In *International Conference on Learning Representations*, 2026b.

Weixiang Zhao, Xingyu Sui, Yulin Hu, Jiahe Guo, Haixiao Liu, Biye Li, Yanyan Zhao, Bing Qin, and Ting Liu. Teaching language models to evolve with users: Dynamic profile modeling for personalized alignment. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 33162–33199. Curran Associates, Inc., 2025. doi: 10.52202/085713-1115. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/2fbc1ea60e215275e823c86338417162-Paper-Conference.pdf.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. Memorybank: Enhancing large language models with long-term memory. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 19724–19731, 2024.

A Limitations

Our evaluation focuses on offline candidate ranking on HorizonBench with two backbone models. This setting supports controlled comparisons under static and evolving preferences, but does not directly establish effectiveness in open-ended generation or live interactions where model responses influence subsequent user feedback. Evaluation on additional benchmarks and interactive user studies would help assess transfer beyond the current setting. PULSE is evaluated with an initial user profile and preceding feedback-bearing interactions. The unseen-user setting therefore examines generalization to users absent from training, rather than personalization without prior user information. Extending PULSE to history-free cold-start settings, through profile initialization and active preference elicitation, remains a direction for future work.

B Related Work

B.1 Long-Term User Profiling and Profile Updating

Long-term LLM personalization commonly represents users through retrieved interaction records, persistent memories, or natural-language profiles. MemoryBank (Zhong et al., 2024) maintains long-term memories and user portraits across repeated interactions, while LaMP (Salemi et al., 2024) and LongLaMP (Kumar et al., 2024) retrieve historical user records to support personalized text generation. For long-term dialogue, LD-Agent (Li et al., 2025) separately maintains event memories and dynamically extracted personas. Other approaches learn compact user representations from lengthy histories: RLPF (Wu et al., 2025) optimizes user summaries using downstream prediction feedback, whereas PLUS (Nam et al., 2026) jointly learns natural-language user summaries and summary-conditioned reward models. These methods establish memory and profile representations as practical interfaces between historical interactions and future responses.

Recent work has further considered how user representations evolve over time. RLPA (Zhao et al., 2025) learns to iteratively infer and refine user profiles through interaction, while PersonaMem (Jiang et al., 2025a) and PersonaMem-v2 (Jiang et al., 2025b) evaluate dynamic profiling and agentic memory under evolving or implicitly expressed preferences. AlpsBench (Xiao et al., 2026) extends evaluation across profile extraction, updating, retrieval, and utilization using real-world dialogues. PGMem (Choi et al., 2026) further connects persona signals to their supporting episodic memories through provenance-aware graph edges. Despite this progress, profile revision is generally performed as a single generation or consolidation operation: the evidence supporting a change and the persistence scope of that change are rarely modeled as separate decisions. PULSE instead factorizes profile updating into reusable-evidence selection, tri-state update attribution, and evidence-grounded profile generation, making local refinements and long-term revisions explicitly traceable to their interaction evidence.

B.2 Personalized Reward Modeling and Profile Utilization

Personalized reward modeling aims to capture individual preferences that are obscured by a single population-level reward function. VPL (Poddar et al., 2024) infers latent user variables from preference histories, PAL (Chen et al., 2025) represents individual preferences as mixtures over shared prototypes, and CoPL (Choi et al., 2025) exploits collaborative user–response structure with mixtures of LoRA experts. LoRe (Bose et al., 2025) instead expresses user-specific rewards as combinations of shared low-rank reward functions. Although these approaches support efficient adaptation and generalization, their personalized evaluation logic remains encoded primarily in latent variables, prototype weights, or model parameters. Natural-language approaches improve interpretability by explicitly describing user preferences. SynthesizeMe (Ryan et al., 2025) induces personas from prior preference judgments, while P-GenRM (Zhang et al., 2026b) derives scenario-specific personas and scoring rubrics. P-CHECK (Seo & Lee, 2026) further generates query-specific checklists and weights criteria according to their discriminative value.

An equally important question is whether stored user information should influence the current response. Zhang et al. (2026a) show that LLMs tend to over-select user attributes as relevant and that incorporating valid attributes does not consistently improve personalized responses. Feng et al. (2026) identify a related gap between recalling a preference and applying it in an appropriate behavioral context, while Song et al. (2026) demonstrate that personalized agents may possess relevant user knowledge without translating it into downstream actions. PURPLE (Du et al., 2026) addresses profile selection by optimizing combinations of user records for downstream generation utility rather than semantic relevance alone. Together, these studies show that accurate storage is insufficient: personalized systems must also control when and how stored information affects response generation or evaluation.

PULSE connects profile updating and profile utilization within a single traceable framework. Unlike methods that derive rewards directly from latent user representations or unverified interaction histories, PULSE constructs request-specific evaluation criteria from an evidence-grounded profile whose statements retain their supporting evidence and update states. These criteria guide a personalized residual that selectively refines a profile-free anchor reward, allowing relevant preferences to affect response ranking without letting contextually irrelevant profile information dominate the decision. PULSE therefore links evidence-attributed profile updating with selective personalized evaluation rather than treating them as independent operations.

C Implementation Details.

We implement the profile updater and personalized reward model using Qwen3-4B-Instruct-2507 and Llama-3.2-3B-Instruct as the backbone. Both modules are adapted with LoRA using a rank of 8, a scaling factor of 16, and a dropout rate of 0.05. LoRA is applied to the attention projections and the feed-forward projection layers. The maximum input length is set to 4,096 tokens, and the profile updater generates at most 512 new tokens.

For each profile update, PULSE uses at most eight recent feedback-bearing interactions. During updater training, we sample four candidate update traces with a temperature of 0.8 and include the unchanged profile as an explicit no-update candidate. The updater is optimized for one epoch using

DPO with $\beta = 0.1$ and a learning rate of 5×10^{-7}. Profile candidates are assessed using look-ahead utility, preference-drift consistency, current-preference gain, evidence-attribution consistency, and a profile-growth penalty, with corresponding coefficients of 1.0, 0.5, 0.5, 0.2, and 0.01. Appendix C.1 defines these components and the preference-pair construction procedure.

The profile-free anchor and personalized residual reward are trained for two epochs using AdamW (Loshchilov & Hutter, 2017) with a learning rate of 2×10^{-5} and a batch size of 1. We first train the anchor to capture general response quality and then keep it fixed while optimizing the criterion-guided residual through the final response-ranking objective. After initialization, we perform one round of updater–reward co-adaptation. We first hold the criterion inducer and reward model fixed to score candidate traces and update the updater adapter with DPO. We then regenerate training profiles, refresh the teacher-generated criterion targets, and update the criterion adapter by supervised fine-tuning. Finally, we regenerate criteria with the updated inducer and refit the residual reward. The pretrained backbones and trained anchor remain fixed throughout this round. At inference time, profiles are updated through greedy decoding.

C.1 Candidate-Trace Utility and Preference-Pair Construction

This appendix specifies the utility used to construct updater preference pairs in Section 3.1. The utility evaluates a complete candidate trace through its downstream ranking effect, evidence attribution, and profile growth. Update states are therefore supervised indirectly through trace preferences, without comparison against reference state labels.

Utility overview. For a valid candidate trace $Z = (P^+, E, z)$, we define

$$U(Z) = F(Z) + 0.5D(Z) + 0.5C(Z) + 0.2A(Z) - 0.01L(Z), \tag{4}$$

where F, D, C, A, and L denote look-ahead utility, preference-drift consistency, current-preference gain, evidence-attribution consistency, and profile-growth penalty, respectively. All labeled response sets used to compute this utility belong to the training split.

Current and look-ahead preference gains. Let r_Z denote the final anchor-plus-residual reward obtained using the candidate trace and its request-specific criteria, and let r_{anc} denote the anchor reward. For a training instance t with candidate set Y_t and preferred response y_t^+, define

$$m_t(r) = r(y_t^+) - \log \sum_{y \in Y_t \setminus \{y_t^+\}} \exp r(y), \tag{5}$$

$$g_t(Z) = m_t(r_Z) - m_t(r_{\mathrm{anc}}). \tag{6}$$

The current-preference term is $C(Z) = g_t(Z)$. Thus, it measures the change in the preferred-versus-rejected reward margin relative to the anchor, rather than the correctness of an update-state label.

To evaluate whether the candidate profile remains useful across requests, we additionally compute

$$F(Z) = \frac{\sum_{k=0}^{H-1} \gamma^k g_{t_k}(Z)}{\sum_{k=0}^{H-1} \gamma^k}, \qquad \gamma = 0.85, \tag{7}$$

where $t_0 = t$ and the remaining probes are up to two subsequent instances from the same user within the training split. The available horizon satisfies $1 \le H \le 3$. The candidate profile is held fixed across these probes, while criteria are induced separately for each request. The look-ahead term therefore includes the current instance; the additional term $C(Z)$ places extra emphasis on immediate preference alignment.

Preference-drift consistency. When the current training instance contains an old-preference distractor y_t^{old}, we compute

$$\begin{aligned} D(Z) = &\big[r_Z(y_t^+) - r_Z(y_t^{\mathrm{old}})\big] \\ &- \big[r_{\mathrm{anc}}(y_t^+) - r_{\mathrm{anc}}(y_t^{\mathrm{old}})\big]. \end{aligned} \tag{8}$$

Otherwise, we set $D(Z) = 0$. This term measures whether personalization increases the margin of the current-preference response over the stale-preference response.

Evidence attribution and profile growth. Let p be the fraction of newly introduced or revised profile statements that cite at least one valid interaction ID from the evidence window. Let q be the fraction of selected evidence IDs that are both valid and cited by those statements. We define

$$A(Z) = \begin{cases} \dfrac{2pq}{p+q}, & p+q>0, \\ 0, & p+q=0. \end{cases} \tag{9}$$

Statements are treated as new or revised when their normalized text differs from all statements in the previous profile. We set $p = 1$ when there are no such statements. For a no-update trace, $q = 1$ if the profile is unchanged and no evidence is selected, and $q = 0$ otherwise. For other traces with no selected evidence, we also set $q = 0$. This score measures structural attribution consistency; it does not independently verify semantic entailment.

To discourage unnecessary profile growth, we use

$$L(Z) = \frac{\max\left(0,\, |\mathrm{ser}(P^+)| - |\mathrm{ser}(P^-)|\right)}{1000}, \tag{10}$$

where $\mathrm{ser}(\cdot)$ denotes profile serialization and length is measured in characters.

Scoring and component scales. The current criterion inducer and reward model compute F, D, and C with their parameters held fixed during candidate scoring. The attribution and growth terms are computed deterministically. The reward-based terms remain in reward-score units, $A(Z) \in [0, 1]$, and $L(Z)$ measures growth in thousands of serialized characters. No additional cross-component normalization is applied; Equation 4 specifies their relative weights.

Preference-pair construction. After filtering invalid traces and removing duplicates, we select the highest-utility trace as the preferred trace. Among the remaining traces, we select the highest-utility alternative whose utility is sufficiently lower. Writing their utilities as U^+ and U^-, respectively, the required gap is

$$U^+ - U^- > \max\left(10^{-5},\, 0.02 \max\left(|U^+|,\, |U^-|,\, 10^{-3}\right)\right). \tag{11}$$

Each update context contributes at most one preference pair. Contexts without a qualifying alternative are skipped. The resulting pairs contain the complete serialized traces and are used for updater DPO training.

D Metric Definitions

Let $\mathcal{Y}$ denote the candidate response set, y^+ the gold response aligned with the user's current preference, and $r(y)$ the final reward assigned to candidate $y \in \mathcal{Y}$. Let $\mathrm{rank}(y^+)$ denote the position of the gold response when all candidates are sorted by their rewards in descending order. For an evolving instance, y^{old} denotes the candidate aligned with the user's previous but no longer applicable preference. We use $\mathcal{D}_{\mathrm{test}}$ to denote the test set and $\mathcal{D}_{\mathrm{old}} \subseteq \mathcal{D}_{\mathrm{test}}$ to denote the subset of evolving instances that contain an old-preference distractor.

Top-1 Accuracy (T1A). T1A measures whether the gold response receives the highest reward among all candidates:

$$\mathrm{T1A} = \frac{1}{|\mathcal{D}_{\mathrm{test}}|} \sum_{(\mathcal{Y}, y^+) \in \mathcal{D}_{\mathrm{test}}} \mathbb{I}\left[r(y^+) > \max_{y \in \mathcal{Y} \setminus \{y^+\}} r(y) \right]. \tag{12}$$

A higher T1A indicates more accurate final response selection.

Mean Reciprocal Rank (MRR). MRR measures the position of the gold response in the complete candidate ranking:

$$\mathrm{MRR} = \frac{1}{|\mathcal{D}_{\mathrm{test}}|} \sum_{(\mathcal{Y}, y^+) \in \mathcal{D}_{\mathrm{test}}} \frac{1}{\mathrm{rank}(y^+)}. \tag{13}$$

MRR assigns a larger value when the gold response appears closer to the top of the ranking. Unlike T1A, which only considers whether the gold response is ranked first, MRR also reflects improvements at lower ranking positions.

Current-over-Old Accuracy (COA). COA evaluates whether a model prioritizes the user's current preference over their previous preference:

$$\mathrm{COA} = \frac{1}{|\mathcal{D}_{\mathrm{old}}|} \sum_{(\mathcal{Y}, y^{+}, y^{\mathrm{old}}) \in \mathcal{D}_{\mathrm{old}}} \mathbb{I}\left[r(y^{+}) > r(y^{\mathrm{old}})\right]. \tag{14}$$

A higher COA indicates a stronger ability to distinguish current preference signals from obsolete ones.

E Case Study

Figure 4 presents an illustrative HorizonBench case in which a user's preferred decision format changes. The previous profile records a preference for explicit numbered options, while the latest interaction asks for a choose-your-own-adventure format. PULSE selects the supporting evidence and revises the relevant profile statement toward branching choices. For the current request, it derives a criterion that distinguishes selectable branches from the outdated numbered menu. Among the candidate responses, B follows the old preference, whereas E offers linked branches. The criterion-guided residual further increases E's advantage over B, while the profile-free anchor assesses general response quality. This case illustrates how evidence-grounded profile updating and selective preference use jointly inform response ranking.

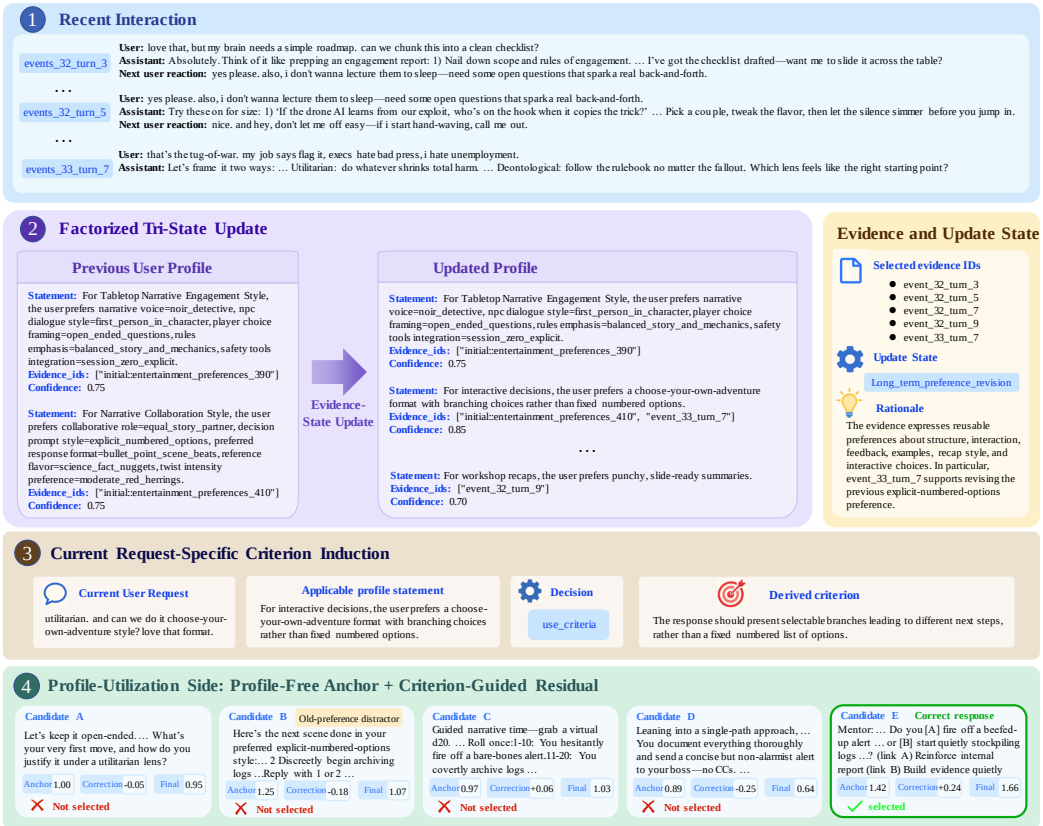
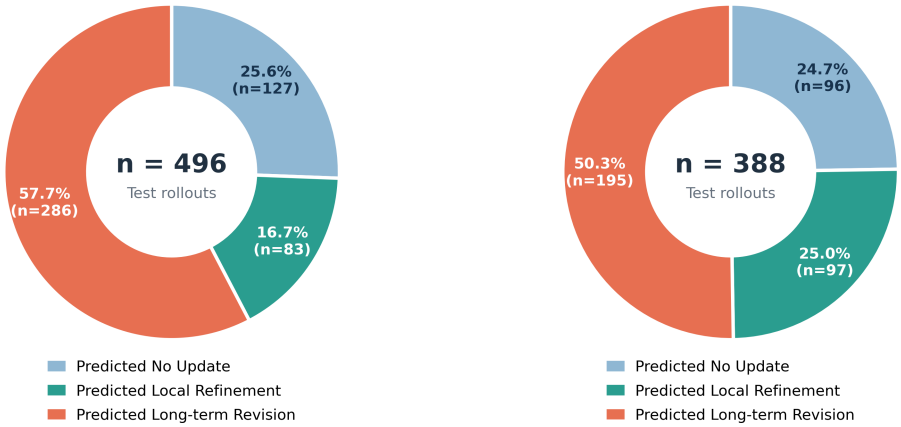


Figure 4: case study.

F Distribution of Profile-Update Decisions

Figure 5 presents the profile-update decisions predicted by PULSE with Qwen3-4B-Instruct-2507 under the seen- and unseen-user settings. PULSE uses all three update states in both settings, rather

than collapsing to a single update action. Long-term revision is the most frequent decision, accounting for 57.7% and 50.3% of the seen and unseen test rollouts, respectively. The proportion of no-update decisions remains similar across the two settings (25.6% vs. 24.7%), whereas local refinement is more frequent for unseen users (25.0% vs. 16.7%).



(a) Seen-user setting.

(b) Unseen-user setting.

Figure 5: Distribution of profile-update decisions predicted by PULSE.

G GPT-Based Annotation Protocol

Evaluation setting. RQ3 and RQ4 use a fixed set of 175 test instances from 18 unseen users with Qwen3-4B-Instruct-2507. The set contains 102 evolving-preference and 73 static-preference instances. Its reference update states comprise 52 *No Update*, 117 *Local Preference Refinement*, and 6 *Long-term Revision* instances. All profile-update methods receive the same previous profile and recent interactions.

Independent annotation and adjudication. We obtain two independent annotations from GPT-5.5 using the same rubric. Method identities are anonymized to avoid quality judgments based on system names. A separate GPT-5.6 Sol pass adjudicates disagreements by examining the instance context. For RQ3, the reference update is finalized before any method output is assessed. The annotation process uses only information contained in each instance.

RQ3 reference construction. For each instance, the annotators select the interactions that contain reusable preference evidence. They then assign one of three update states and construct the reference profile. *No Update* means that no reusable change is supported. *Local Preference Refinement* captures a reusable but context-bound preference. *Long-term Revision* represents the addition, removal, replacement, or substantial revision of a stable preference.

Statement assessment. Each predicted statement is matched to at most one reference statement, and vice versa. A match requires agreement in both preference value and scope. Topical similarity alone is insufficient. Statements copied from the previous profile require no new evidence. A newly introduced or materially revised statement must cite interactions that directly support its content. It is marked unsupported if its citations are absent, irrelevant, or support only a narrower claim.

RQ3 metrics. ES-F1 measures reusable-evidence selection using micro-averaged F1. US-F1 measures update-state attribution using macro-F1 over the three states.

A statement is treated as unchanged when its normalized text appears in the previous profile. All other statements are treated as changed. Let ΔG_i and ΔP_i be the reference and predicted changed-

statement sets for instance i, and let M_i be the number of one-to-one semantic matches between them. CS-F1 is defined as

$$\text{CS-F1} = \frac{1}{|\mathcal{D}_{\text{upd}}|} \sum_{i \in \mathcal{D}_{\text{upd}}} \frac{2M_i}{|\Delta G_i| + |\Delta P_i|}, \tag{15}$$

where $\mathcal{D}_{\text{upd}}$ contains instances whose reference state is not *No Update*.

ESR is the proportion of predicted changed statements that are supported by their cited interactions. An uncited changed statement is counted as unsupported. ESR is undefined for the NO UPDATE baseline because it produces no changed statements.

RQ4 criterion assessment. RQ4 uses the same instances as RQ3. The annotators receive the updated profile, current request, generated criteria, and the source statement of each criterion. Candidate labels and ranking scores are hidden during annotation.

Each request is first labeled according to whether personalization is warranted. Every generated criterion is then assessed for source grounding, request applicability, and nonredundancy with the explicit request. A criterion is valid only if it satisfies all three conditions. For a request that warrants personalization, the criterion decision is correct when PULSE produces a nonempty set containing only valid criteria. Otherwise, the decision is correct when PULSE produces no criterion. Decision Accuracy is reported separately for the two request types.

Residual effects. We compute residual effects from the saved anchor and final rankings. A *Helped* case occurs when the anchor selects an incorrect response but the final reward selects the reference response. A *Hurt* case occurs when the anchor is correct but the final reward is not. Table 3 reports the number of cases in each category, separated by whether personalization is warranted.

H PULSE PROMPT TEMPLATES

We present the prompt templates used by PULSE for profile updating, criterion construction, and reward scoring. Angle-bracketed text denotes instance-specific content. The profile updater and criterion inducer receive system and user messages, whereas the two reward templates are text inputs to scalar scoring models rather than requests for generated judgments.

Training the criterion inducer. For each training instance, we obtain an evidence-state profile from the updater rollout and combine it with the current request and labeled candidate responses to construct the teacher input. The teacher is the corresponding pretrained instruction backbone—Qwen3-4B-Instruct-2507 or Llama-3.2-3B-Instruct—with LoRA adapters disabled. It generates a JSON decision containing either source-attributed criteria or `no_personalization`. Outputs are checked for valid structure and source references; malformed outputs receive one format-repair retry, and remaining invalid targets are discarded. Instances without eligible source statements directly return `no_personalization` and are excluded from criterion-adapter training.

The criterion inducer shares the updater's frozen backbone but uses a separate LoRA adapter. Each supervised example pairs the response-free inference prompt in Figure 7 with the teacher-generated decision $\hat{d}$. We minimize the completion-token cross-entropy:

$$\mathcal{L}_{\text{crit}} = -\mathbb{E}_{(Z,x,\hat{d})} \left[\frac{1}{|\hat{d}|} \sum_{j=1}^{|\hat{d}|} \log p_\phi \left(\hat{d}_j \mid Z, x, \hat{d}_{<j} \right) \right], \tag{16}$$

where all prompt tokens are masked from the loss. Only the criterion adapter is updated, using the LoRA configuration in Appendix C and AdamW for two epochs with a learning rate of 2×10^{-5}. When both decision types are present, valid source-bearing `no_personalization` targets are oversampled, if needed, to comprise at least 25% of the training examples in each epoch. Old-preference distractors are marked as temporal hard negatives in the teacher input; this annotation guides target construction and is absent from the student input. The learned criterion inducer is subsequently used for both residual-reward training and inference.

Factorized Tri-State Profile Update Prompt

System message

You are a dynamic user-profile updater.
Generate the complete next profile, not an edit instruction.
Model the update as a learned latent trace with three ordered fields.
First select the Evidence IDs that contain reusable profile information.
Then choose exactly one update_state:
- no_update: the evidence does not justify changing reusable profile preferences.
- local_preference_refinement: the evidence refines a situational or context-bound response preference.
- long_term_preference_revision: the evidence supports revising a stable preference or replacing stale profile information.

Finally generate the complete next profile.
You may retain, add, merge, rewrite, or delete statements when supported by the selected evidence.
If there is no reusable profile evidence, output selected_evidence_ids=[], update_state=no_update, and reproduce the previous profile.
Every newly added or materially revised profile statement must cite one or more Evidence IDs from selected_evidence_ids.
Return exactly one JSON object with this shape:

```
{"selected_evidence_ids":["event_1_turn_3"],"update_state":"local_preference_refinement",
"profile":[{"statement":"...","evidence_ids":["event_1_turn_3"],"confidence":0.0}]}
```

For coverage, select all Evidence IDs that contribute distinct reusable response preferences, not merely one valid example.
Represent each selected preference as an atomic, observable response property; do not merge unrelated content, style, structure, reasoning, citation, format, or numeric-choice attributes into one broad statement.
Preserve concrete preference values such as a requested citation style, threshold, format, tone, or interaction pattern instead of replacing them with a topical or motivational summary.
When update_state is not no_update, preserve every distinct supported preference from the selected evidence in the complete profile and cite its Evidence ID.

User message

[Previous Profile JSON]

<previous_profile_json>

[Recent Chronological Evidence]

[Evidence ID: event_<event_idx>_turn_<turn_idx>]
Date: <date>
User request: <past_user_request>
Assistant response: <past_assistant_response>
Next user reaction: <user_feedback>
<additional evidence blocks, if present>

[Factorized Profile Update JSON]

Figure 6: Factorized Tri-State Profile Update Prompt.

Criterion Construction Prompts

Training-only teacher prompt

System message

You supervise a personalized-criterion policy from response contrasts.
The chosen and rejected responses are training-only evidence. Produce the smallest set of profile-grounded, query-applicable, atomic and observable criteria that explain the personalized chosen-versus-rejected differences.
A response marked is_old_preference is a temporal hard negative, so no criterion may favor that stale preference.
The output must be exactly one of these JSON objects:

```
{"decision":"no_personalization","criteria":[]}{"decision":"use_criteria","criteria":[{"sourc
e_statement_id":"p1","criterion":"The response should ..."}]}
```

Only the keys shown above are allowed.
Use only supplied profile statements. Do not encode answer-specific words, response IDs, request facts, or the fact that an answer was chosen into a criterion. Return no_personalization when no supplied profile statement explains an observable chosen-versus-rejected difference.
Return JSON only.

User message

```
[Profile Update State]
<update_state>
[Attributed Profile Statements]
<source_statements_json>
[Current User Request]
<current_request>
[Training-only Chosen Response]
<chosen_response>
[Training-only Rejected Responses]
<rejected_responses_json>
[Criterion Decision JSON]
```

Inference-time prompt

System message

You induce query-specific personalized evaluation criteria.
Input contains only evidence-attributed profile statements, their current/inherited role, the learned profile-update state, and the current request.
The output must be exactly one of these JSON objects:

```
{"decision":"no_personalization","criteria":[]}{"decision":"use_criteria","criteria":[{"source
_statement_id":"p1","criterion":"The response should ..."}]}
```

Only the keys shown above are allowed.
Requirements:

- Select only statements that would meaningfully change how this request should be answered.
- Return no_personalization when none of the supplied statements changes an observable property of a valid response to this request; topical similarity alone is not sufficient.
- Rewrite selected preference attributes into observable response properties.
- Each criterion must test exactly one content, style, structure, or reasoning property.
- Split a multi-attribute statement into separate atomic criteria when more than one attribute is applicable.
- Do not include generic quality requirements unless the profile explicitly personalizes them.
- Do not copy request facts into a criterion and do not invent a preference. Cite exactly one supplied source_statement_id for every criterion.
- Produce the smallest sufficient set; criterion order has no meaning.
- Return JSON only.

User message

```
[Profile Update State]
<update_state>
[Attributed Profile Statements]
<source_statements_json>
[Current User Request]
<current_request>
[Criterion Decision JSON]
```

Figure 7: Criterion Construction Prompt.

Reward Scoring Prompts

Profile-free anchor reward

You are a personalized reward model.
Score whether the candidate response fits the current request and user profile.

```
[User Profile]
No reliable response preferences have been inferred yet.
[Current User Request]
<current_request>
[Candidate Assistant Response]
<candidate_response>
[Reward]
```

Criterion-guided residual reward

You are a personalized criterion-set reward model.
Score only the valid personalized contribution of the criterion set for this request. First determine whether the set describes an applicable, observable response distinction. An inapplicable set must produce a neutral correction; when applicable, evaluate its criteria jointly. Do not retrieve profile statements or add criteria.

```
[Current User Request]
<current_request>
[Personalized Criterion Set]
<criterion_strings_json>
[Candidate Assistant Response]
<candidate_response>
[Criterion Satisfaction Reward]
```

Figure 8: Reward Scoring Prompt.