# BRIDGING EXPLORATION AND PERSONA: INFORMATION-DIRECTED DUELING FOR TEST-TIME AGENT PERSONALIZATION

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Effective test-time personalization requires LLM agents to adapt to query-dependent user preferences from lightweight interaction. We study persona selection with binary comparisons between candidate responses as the online signal. We present ID-TAP, a scalable, IDS-inspired neural dueling-bandit framework for large discrete persona spaces. ID-TAP combines a query-conditioned neural utility model with a lightweight Bradley–Terry explorer, and transfers comparison evidence across related queries through similarity-weighted warm starts. The neural model shares statistical strength across queries while preserving query-specific scores, and efficient BT updates score hypothetical challengers. Across seven user-conditioned generation benchmarks under reference- and model-based preference simulators, ID-TAP achieves the strongest aggregate performance among the evaluated baselines within a fixed ten-round comparison budget. These results support information-directed challenger selection as an effective design for test-time persona adaptation.

## 1 INTRODUCTION

Large language model (LLM)-based agents support tasks ranging from creative writing to coding assistance (Huang et al., 2023; Li et al., 2024a). Their behavior, including response style and persona, is strongly influenced by the instructions supplied at inference time. Prompt design and automatic instruction optimization improve task-level behavior (Wei et al., 2022; Zhou et al., 2022); personalization extends this objective toward responses suited to an individual user (Salemi et al., 2024).

A natural control interface is the agent's *persona*: a prompt-level description of the behavior and style the agent should adopt. Persona selection differs from task-level prompt optimization in three ways. First, personalization is **contextual**: the persona most useful for one request may be less suitable for another (e.g., a "creative collaborator" for brainstorming versus a "rigorous critic" for editing) (Salemi et al., 2024; Zhang et al., 2025a). Second, user preferences can be **implicit and noisy**: interaction histories provide partial evidence about relevant tendencies (Richardson et al., 2023). Third, candidate persona descriptions can be **semantically dense**. Closely related descriptions (e.g., "stern tutor" and "strict mentor") may induce meaningfully different responses despite high embedding similarity.

These properties motivate **test-time adaptation (TTA)** over a fixed persona pool. Prompt-level adaptation preserves a shared backbone while incorporating feedback about the current request. PersonaAgent (Zhang et al., 2025a) uses expressive linguistic feedback, while **preference feedback** reduces an interaction to choosing between two candidate responses, a supervision format also used in RLHF (Ouyang et al., 2022).

Recent approaches such as POHF (Lin et al., 2024) and PDO (Wu et al., 2025) apply bandit exploration to preference-driven prompt optimization. In the persona pools evaluated here, their UCB- and Thompson-sampling-based components revisit challengers more often and reach lower aggregate performance than our information-directed rule (Sec. 4).

Building on information-directed and dueling-bandit methods, we develop ID-TAP (**I**nformation-Directed **D**ueling for **T**est-Time **A**gent **P**ersonalization), a scalable challenger-selection frame-



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.

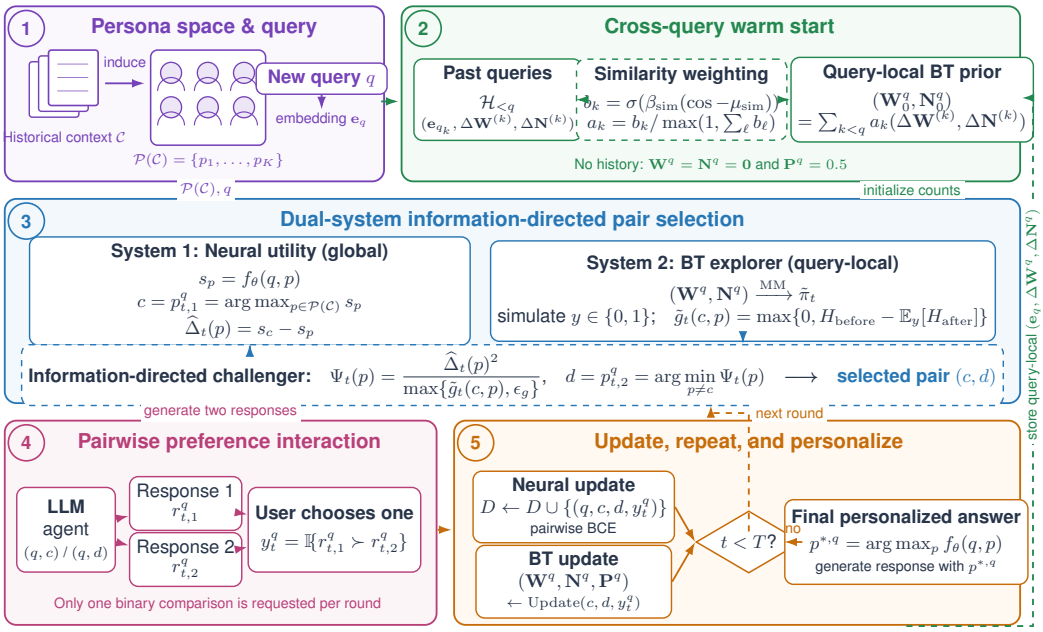



Figure 1: **Overview of ID-TAP.** Historical context induces a shared persona space and each new query inherits similarity-weighted BT statistics from prior queries (1–2). At every interaction round, the neural utility model selects the champion and estimates utility gaps, while the query-local BT explorer computes a plug-in entropy-reduction proxy; their acquisition ratio selects the challenger (3). The agent generates the resulting response pair, obtains one binary preference, and updates both systems (4–5). After $T$ rounds, the highest-scoring persona produces the personalized response, and the query-local BT increments are retained for later warm starts.

work that uses pairwise preferences as its online signal. Its technical focus is the joint design required by black-box personalization over large persona pools: query-conditioned utility estimation, lightweight hypothetical BT updates, and transfer across related queries. Specifically, a query-conditioned neural utility model estimates utility gaps, while a lightweight statistical explorer evaluates hypothetical comparison outcomes through an IDS-inspired entropy-reduction proxy. This separation supports efficient challenger scoring with one neural update per observed comparison, and similarity-weighted warm starts reuse preference evidence across queries. Under a fixed comparison budget, the resulting method attains the strongest aggregate result among the evaluated baselines.

Our contributions are summarized as follows:

**Scalable Challenger Acquisition.** We operationalize an IDS-inspired acquisition criterion for large discrete persona pools by combining neural utility gaps with a lightweight BT entropy-reduction proxy, enabling efficient scoring across candidate challengers.

**Cross-Query Adaptation.** We introduce a similarity-weighted warm start that reuses comparison evidence across related requests while retaining query-conditioned utility prediction.

**Empirical Evaluation.** Across seven benchmarks and two offline preference simulators, ID-TAP achieves the strongest aggregate result among the compared methods under the same ten-round budget. On the aggregate generation-task trajectory, it overtakes the strongest compared baseline after approximately **4 simulated-feedback rounds**, suggesting favorable feedback-round efficiency in this setting.

# 2 PROBLEM FORMULATION

We formalize persona selection as a contextual, query-level problem with an offline construction phase and an online test-time adaptation phase.

## 2.1 PERSONA AND CONTEXTUAL SPACE

For a personalized generation task, a *persona* $p$ is a prompt-level description that summarizes relevant traits, stylistic habits, and task preferences inferred from the user's historical interaction context $\mathcal{C}$. The offline procedure constructs a finite persona space $\mathcal{P}(\mathcal{C}) = \{p_1, \ldots, p_K\}$ from this history. The candidates capture different emphases that remain grounded in $\mathcal{C}$. Because requests from the same user can differ in intent and complexity, their utilities are query-dependent rather than governed by a single universally best candidate.

## 2.2 OFFLINE–ONLINE FRAMEWORK

We use the following offline–online protocol:

**Offline Phase (Persona Induction).** Historical context $\mathcal{C}$ is used to construct the shared candidate set $\mathcal{P}(\mathcal{C})$. The candidate strings are fixed during online adaptation and can be reused across the user's subsequent queries.

**Online Phase (Query-Specific Adaptation).** At test time, the agent receives a query $q$. Let $u(q, p)$ denote the unknown expected utility of the response generated under persona $p$. The query-conditioned optimal candidate is

$$p^{*,q} = \arg \max_{p \in \mathcal{P}(\mathcal{C})} u(q, p). \tag{1}$$

The neural score $f_\theta(q, p)$ introduced in Section 3 estimates this latent utility from observed comparisons.

## 2.3 TEST-TIME ADAPTATION WITH PREFERENCE FEEDBACK

ID-TAP learns estimates of the latent utility online while keeping the finite candidate personas fixed. We assume that user preferences are locally stable over short interaction horizons. Test-time adaptation therefore operates within the pre-constructed persona space $\mathcal{P}(\mathcal{C})$, with the historical context $\mathcal{C}$ fixed throughout the session.

To identify a strong persona from lightweight interaction, we adopt a dueling-bandit interface in which the agent learns from pairwise preference feedback. At interaction step $t$, the algorithm selects a champion persona and a challenger persona, then generates the corresponding response pair $(r_t^1, r_t^2)$. The champion exploits the neural model's current estimate, whereas the challenger is chosen by the IDS-inspired acquisition rule in Section 3. The user returns a binary comparison $y_t = \mathbb{I}\{r_t^1 \succ r_t^2\}$, which updates both utility estimates and pairwise statistics. Given an interaction budget $T$, the objective is to return a persona $\hat{p}_T^q$ with high query-conditioned utility from these binary comparisons.

# 3 METHODOLOGY

ID-TAP uses an **IDS-inspired acquisition** strategy for neural dueling bandits in a personalized, test-time interaction setting. Before test time, we assume that the historical context set $\mathcal{C}$ and the corresponding persona space $\mathcal{P}(\mathcal{C})$ have been constructed offline. Inspired by **Information-Directed Sampling (IDS)**, our method balances exploitation and exploration by selecting comparison pairs that are both performance-relevant and uncertainty-reducing. Specifically, the challenger is chosen by relating its estimated utility gap to a plug-in entropy-reduction proxy computed from the BT explorer.

**Algorithm 1** ID-TAP

**Initialize** Neural network $f_\theta$, persona space $\mathcal{P}(\mathcal{C})$
**for** $q = 1, 2, \ldots, N$ **do**
  Initialize $(\mathbf{W}^q, \mathbf{N}^q, \mathbf{P}^q)$ by Algorithm 2; set $(\Delta\mathbf{W}^q, \Delta\mathbf{N}^q) \leftarrow (\mathbf{0}, \mathbf{0})$
  **for** $t = 1, 2, \ldots, T$ **do**
    $p^q_{t,1} \leftarrow \arg\max_{p \in \mathcal{P}(\mathcal{C})} f_\theta(q, p)$
    $p^q_{t,2} \leftarrow \arg\min_{p \in \mathcal{P}(\mathcal{C}) \setminus \{p^q_{t,1}\}} \frac{\hat{\Delta}_t(p)^2}{\max\{\tilde{g}_t(p^q_{t,1}, p), \epsilon_g\}}$
    Generate response pair $\{r^q_{t,1}, r^q_{t,2}\}$
    Observe preference feedback $y^q_t \leftarrow \mathbb{I}\{r^q_{t,1} \succ r^q_{t,2}\}$
    Train $f_\theta$ with $D^q_t = \bigcup_{k<q} D^{q_k}_T \cup \{(q, p^q_{s,1}, p^q_{s,2}, y^q_s)\}_{s \le t}$
    Update $(\mathbf{W}^q, \mathbf{N}^q, \mathbf{P}^q)$ and local increments $(\Delta\mathbf{W}^q, \Delta\mathbf{N}^q)$
  **end for**
  Store $(q, \mathbf{e}_q, \Delta\mathbf{W}^q, \Delta\mathbf{N}^q)$; output $\hat{p}^q_T \leftarrow \arg\max_p f_\theta(q, p)$
**end for**

### 3.1 Algorithm Overview

At test time, a user sequentially queries an AI agent with personalized requests. For each query $q$, the agent interacts with the user for a fixed budget of $T$ rounds, collecting pairwise feedback to adapt persona selection to the query-conditioned preference. The neural parameters persist across a user's queries; the BT state is initialized by Algorithm 2 and then updated locally. For the first query, the BT counts are zero and its pairwise probabilities are neutral. We use a fixed candidate-index rule to resolve exact ties in champion or challenger scores.

We model the utility function with a neural network. At each round, the agent first selects an exploitation arm

$$p^q_{t,1} = \arg\max_{p \in \mathcal{P}(\mathcal{C})} f_\theta(q, p), \tag{2}$$

corresponding to the currently estimated best persona.

To efficiently explore alternative personas and reduce uncertainty, the second arm $p^q_{t,2}$ is selected using an information-directed criterion. Concretely, we choose the challenger to minimize the acquisition ratio:

$$p^q_{t,2} = \arg\min_{p \in \mathcal{P}(\mathcal{C}) \setminus \{p^q_{t,1}\}} \frac{\hat{\Delta}_t(p)^2}{\max\{\tilde{g}_t(p^q_{t,1}, p), \epsilon_g\}}, \tag{3}$$

where $\hat{\Delta}_t(p) = f_\theta(q, p^q_{t,1}) - f_\theta(q, p)$ is the estimated utility gap, $\tilde{g}_t$ is the BT-based entropy-reduction proxy defined below, and $\epsilon_g > 0$ prevents division by zero. The ratio favors challengers with a small predicted utility loss and a large estimated reduction in ranking uncertainty.

The user then provides preference feedback in the form of a binary comparison outcome. All collected pairwise interactions across queries and rounds are aggregated and used to update the neural utility model. After round $T$, the highest-scoring persona $\hat{p}^q_T$ conditions the final generated response.

### 3.2 IDS-Inspired Challenger Selection

Building on information-directed methods for bandit and preference feedback (Russo & Van Roy, 2018; Kirschner et al., 2020; Kirschner & Krause, 2021; Pásztor et al., 2024), **ID-TAP** adapts the information-ratio principle to persona-level pair selection. We fix $p^q_{t,1}$ as the current greedy champion and select a challenger $p^q_{t,2}$ by minimizing the acquisition ratio.

**Plug-in Entropy-Reduction Proxy.** The BT solver produces strengths $\hat{\theta}^q_t$, which we map to the normalized plug-in score distribution $\tilde{\pi}^q_t(k) = \mathrm{softmax}(\hat{\theta}^q_t/\tau)_k$. For a champion–challenger pair

$(p^q_{t,1}, p)$, we define

$$\tilde{g}_t(p^q_{t,1}, p) = \max\left\{0, H(\tilde{\pi}^q_t) - \sum_{y\in\{0,1\}} \hat{P}_t(y \mid p^q_{t,1}, p) H(\tilde{\pi}^{q,(y)}_t)\right\}, \tag{4}$$

where $H(\tilde{\pi}) = -\sum_k \tilde{\pi}(k)\log\tilde{\pi}(k)$, $\hat{P}_t$ is the BT predictive probability, and $\tilde{\pi}^{q,(y)}_t$ is obtained by adding hypothetical outcome $y$ and re-estimating the BT strengths. This IDS-inspired plug-in score ranks pairs by their predicted reduction in the dispersion of normalized BT strengths. Reproducible implementation details are provided in Appendix A.

### 3.3 Dual-System Architecture for Real-Time Efficiency

Equation (4) evaluates two hypothetical outcomes for each candidate, yielding $2(K-1)$ BT re-estimations per interaction round. The lightweight BT solver makes this candidate-wide scoring practical at large $K$, while the neural utility model performs one global update after each observed comparison. We therefore decouple global utility learning from local uncertainty estimation:

- **System 1: Neural Utility Model ($f_\theta$)** This system serves as the shared utility estimator. It maps queries and persona embeddings to utility scores $s_p$ and provides the gap estimates $\hat{\Delta}_t(p)$ from the interaction history $D$ accumulated across queries. It is updated once per interaction round after observing feedback.
- **System 2: Statistical Explorer (BT-MLE)**
  This system acts as the query-local explorer. It maintains a lightweight Bradley–Terry (BT) model using comparison counts $\{w_{ij}, n_{ij}\}$. Its minorization–maximization (MM) solver can simulate the $2(K-1)$ hypothetical outcomes and recompute the normalized strength distribution efficiently.

**Pairwise Probabilities.** The BT explorer applies symmetric Beta smoothing to observed pairs and assigns neutral probability $P_{ij} = 0.5$ to unobserved pairs. Appendix A gives the count updates, probability estimator, and MM solver.

**Design Motivation.** The dual-system design combines complementary estimation properties. System 1 (the neural utility model $f_\theta$) is trained with pairwise BCE, corresponding to the negative BTL log-likelihood used by System 2 (BT-MLE). System 1 selects the champion and estimates utility gaps through shared representations, while System 2 supplies the challenger-side entropy-reduction proxy from pairwise outcomes. Consequently, $\Psi_t(p) = \hat{\Delta}_t(p)^2 / \max\{\tilde{g}_t(p^q_{t,1}, p), \epsilon_g\}$ selects challengers that are both promising and informative. Existing neural-dueling, Bradley–Terry estimation, and classical M-estimation analyses motivate these components (Verma et al., 2025; Hunter, 2004; Van der Vaart, 2000; Shah et al., 2015); our empirical study evaluates their coupled use with cross-query warm starts.

**Cross-Query Knowledge Transfer.** For a new query $q$, ID-TAP warm-starts the BT system from the query-local count increments $(\Delta\mathbf{W}^{(k)}, \Delta\mathbf{N}^{(k)})$ collected on earlier queries. Let $b_k$ be the raw similarity score and normalize it as

$$\begin{aligned} b_k &= \sigma(\beta_{\text{sim}}(\text{sim}(q, q_k) - \mu_{\text{sim}}))\,, \\ a_k &= \frac{b_k}{\max\{1, \sum_{\ell<q} b_\ell\}}, \qquad \sum_{k<q} a_k \le 1, \\ (\mathbf{W}^q_0, \mathbf{N}^q_0) &= \sum_{k<q} a_k(\Delta\mathbf{W}^{(k)}, \Delta\mathbf{N}^{(k)}). \end{aligned} \tag{5}$$

The history stores only the counts generated locally on each query $q_k$, so each local increment enters any later query's initialization at most once. The normalized weights cap the total inherited count mass at one query-equivalent. The inherited counts are converted to $\mathbf{P}^q_0$ using Eq. (8); the neural model separately learns from the interaction history $D$.

Table 1: Performance across seven benchmarks under two proxy-feedback mechanisms. R.L. is per-query normalized ROUGE-L. W.R. is the final-response win rate against a naive response and is reported as mean±SEM across users. Dataset results aggregate per-user means from three independent runs; the last row is the macro-average of dataset means. **Bold** marks the best result and underlining marks the second best.

| Dataset | PersonaAgent | | POHF | | PDO Sampling | | ID-TAP | |
|---|---|---|---|---|---|---|---|---|
| | R.L | W.R. | R.L | W.R. | R.L | W.R. | R.L | W.R. |
| LaMP-4 | 0.607 | 70.6±8.6 | 0.659 | 68.3±6.7 | 0.660 | 69.4±7.5 | **0.781** | **73.0±5.5** |
| LaMP-5 | 0.723 | 59.7±1.8 | 0.822 | 67.5±1.0 | 0.817 | 68.7±1.4 | **0.924** | **77.2±0.4** |
| LongLaMP-2 | 0.887 | **88.8±1.3** | 0.880 | 79.5±1.3 | 0.865 | 83.2±0.7 | **0.939** | 84.7±2.0 |
| LongLaMP-3 | 0.882 | 82.3±3.0 | 0.914 | 80.3±2.8 | 0.912 | 77.2±3.5 | **0.964** | **85.7±2.1** |
| LongLaMP-4 | 0.864 | 85.2±1.3 | 0.868 | 84.9±1.6 | 0.872 | 84.0±0.7 | **0.962** | **87.9±0.2** |
| PrefEval | 0.774 | 59.2±4.6 | 0.943 | 81.7±5.8 | 0.886 | 83.3±3.0 | **0.956** | **85.8±1.7** |
| UltraChat | 0.369 | 86.7±2.2 | 0.875 | 90.0±1.7 | 0.845 | 89.2±2.9 | **0.907** | **90.8±1.7** |
| **Average** | 0.729 | 76.1 | 0.852 | 78.9 | 0.837 | 79.3 | **0.919** | **83.6** |

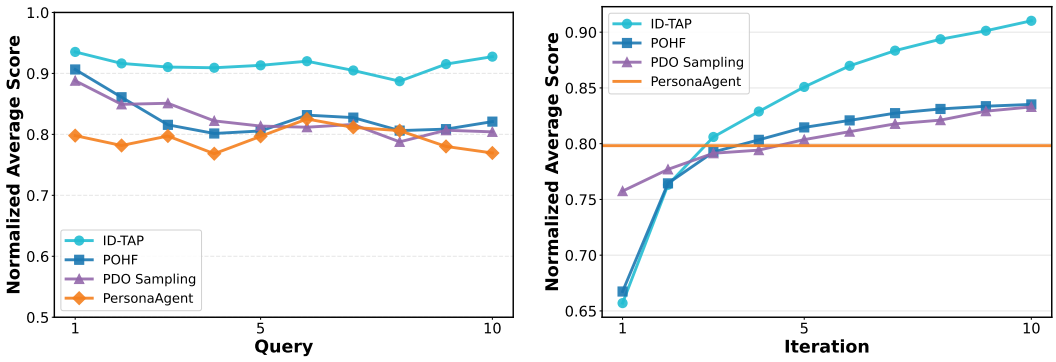



Figure 2: Performance averaged over LaMP-4, LaMP-5, LongLaMP-2, LongLaMP-3, and LongLaMP-4. Left: query-wise normalized ROUGE-L. Right: optimization over ten simulated-preference rounds.

# 4 Experiments

## 4.1 Experimental Setup

**Configuration.** We adopt *DeepSeek-V3.2* (DeepSeek-AI, 2025) as the backbone LLM for the agent and *Qwen3-Embedding-4B* (Zhang et al., 2025b) as our embedding model. The main experiments use $K = 400$ candidate personas per user. For each user, every experimental configuration is run independently three times, and the arithmetic mean of those runs is treated as that user's result. Dataset-level means, standard deviations, and standard errors are computed across these per-user results. Geometric statistics are computed for each user's persona pool before aggregation. $N_{\text{eval}}$ denotes query-level evaluation instances, whereas $n_{\text{user}}$ denotes sampled users in the auxiliary sensitivity and runtime analyses.

**Datasets.** Following *PersonaAgent*, we evaluate on **LaMP** (Salemi et al., 2024). We also include **LongLaMP** (Kumar et al., 2024), **PrefEval** (Zhao et al., 2025), and **UltraChat** (Ding et al., 2023), covering short- and long-form personalized generation and conversational settings. Dataset details and statistics are provided in Appendix B.

## 4.2 Baselines and Shared Evaluation Protocol

We compare ID-TAP against a generative personalization baseline and two online preference-learning baselines.

**Generative Baseline.** *PersonaAgent* (Zhang et al., 2025a) performs personalization by synthesizing user-specific information into prompts.

**Online Learning Baselines.** Since our setting involves iterative preference learning, we compare against two dueling-bandit acquisition rules: *POHF*, which uses a neural UCB-style rule, and the *PDO sampling component*, i.e., Double Thompson Sampling (D-TS) (Wu & Liu, 2016; Wu et al., 2025). Full PDO combines sampling with prompt mutation and pruning. We instantiate its D-TS sampling component on the shared fixed persona pool, yielding a matched comparison of pair selection under the same search space and feedback budget.

**Shared Evaluation Protocol.** PersonaAgent follows its reported configuration. All bandit methods share the same candidate personas, response generator, binary-feedback interface, and ten-round budget for each query. Within this common protocol, PDO Sampling represents its acquisition component and ID-TAP represents the proposed acquisition design.

### 4.3 Preference Feedback Mechanisms

ID-TAP's online update consumes one binary outcome for each displayed response pair. In deployment, the user supplies this outcome after viewing the pair. For repeatable offline experiments, the held-out target instantiates a *proxy user* that returns the same type of outcome after each algorithm-selected comparison. Following POHF's offline interaction protocol (Lin et al., 2024), persona construction, response generation, and arm selection use the history, query, and observed binary outcomes; after pair selection, the proxy user uses the held-out target to produce the next outcome. This protocol reproduces test-time information flow while standardizing the preference source across methods for $T = 10$ rounds.

For final W.R. evaluation, each method's final response is paired with a *naive response*: an answer generated by the same backbone directly from the query and user history or context, without persona selection or test-time preference optimization. The pair order is randomized before evaluation, and W.R. is the percentage of comparisons in which the method response is preferred to the naive response.

**ROUGE-L-Based Simulation.** For each candidate response $r^q_{t,i}$, the proxy user computes the reference-grounded utility $s^q_{t,i} = \mathrm{ROUGE\text{-}L}(r^q_{t,i}, r^q_{gt})$. Following the offline preference-simulation protocol of POHF (Lin et al., 2024), we sample the binary outcome using the Bradley–Terry–Luce model:

$$P(r^q_{t,1} \succ r^q_{t,2}) = \frac{\exp(s^q_{t,1})}{\exp(s^q_{t,1}) + \exp(s^q_{t,2})}. \tag{6}$$

**Reference-Grounded Holistic Preference Proxy.** An LLM judge receives the query, a user-specific reference, and an anonymized response pair. For every comparison, the two responses are randomly assigned to the Response 1 and Response 2 positions, and the judge's selection is mapped back to the corresponding personas. The judge selects the response that better realizes the intended output as a whole, jointly considering task fulfillment, semantic intent, organization, coherence, and the tone and style expressed by the reference. The reference supplies a task- and user-conditioned example, and the judge balances semantic realization with organization and style. We refer to this signal as a reference-grounded holistic preference proxy. In both settings, the optimizer's adaptation signal is the resulting binary comparison, while the proxy user retains the reference and its derived score.

### 4.4 Main Results and Analysis

**Overall Performance.** Table 1 summarizes final performance across the seven datasets. Under ROUGE-L-based preference simulation, R.L. values are normalized per query by the maximum value among the evaluated methods; under LLM-based preference simulation, W.R. measures how often the method's final response is preferred to the naive response and is reported as mean$\pm$SEM across users. ID-TAP is numerically best in 13 of the 14 dataset–metric cells and has the strongest average in both columns; the exception is LongLaMP-2 W.R., where PersonaAgent is higher. The raw ROUGE-L table yields the same first-place ordering for ID-TAP on all seven datasets, with

a cross-dataset average of 0.2228 versus 0.2068 for the strongest baseline. Appendix B.4 reports unnormalized ROUGE-L as mean±SEM together with evaluation-set sizes.

**Query-Level Performance on Generation Tasks.** The left panel of Figure 2 shows query-wise ROUGE-L averaged over the five generation tasks LaMP-4, LaMP-5, LongLaMP-2, LongLaMP-3, and LongLaMP-4. Separate plots for every task, including PrefEval and UltraChat, are provided in Appendix G.1.

**Feedback-Round Efficiency and Dynamics.** We analyze the optimization dynamics under ROUGE-L-based preference simulation in the right panel of Figure 2. The plot illustrates iteration-level performance averaged across the five LaMP and LongLaMP generation tasks shown in the figure. ID-TAP exceeds the compared baselines after approximately **4 simulated-feedback rounds** on this aggregate curve, indicating faster optimization under the shared evaluation protocol. Detailed iteration-level trajectories for all seven tasks, including PrefEval and UltraChat, are provided in Appendix G.2.

### 4.5 Matched Component Ablations

All variants independently produce their own test-time trajectories under the same persona pool, query order, comparison budget, model configuration, and evaluation protocol. *Random Challenger* retains the neural champion, both updates, cross-query states, and neural final selector, changing only the challenger to a uniform non-champion draw. *ID-TAP-NH* reinitializes the neural model and neutralizes the BT statistics for every query, whereas *ID-TAP-NBT* retains neural history but neutralizes BT statistics; hence NH versus NBT isolates neural-history transfer and NBT versus Full isolates the BT warm start. *Random Response* and *Random Acquisition + Neural Final* are additional diagnostics of random data collection and output selection.

Table 2: Normalized ROUGE-L for matched component ablations, aggregated from per-user means over three independent runs. NH removes neural and BT history; NBT retains neural history but removes BT transfer.

| Method | LongLaMP-3 | LongLaMP-4 | PrefEval |
|---|---|---|---|
| Random Response | 0.601 | 0.497 | 0.448 |
| Random Acquisition + Neural Final | 0.615 | 0.516 | 0.507 |
| ID-TAP-NH | 0.764 | 0.647 | 0.629 |
| ID-TAP-NBT | 0.774 | 0.657 | 0.629 |
| Random Challenger | 0.778 | 0.644 | 0.618 |
| **Full ID-TAP** | **0.783** | **0.664** | **0.629** |

Full ID-TAP exceeds Random Challenger by 0.005, 0.020, and 0.011, supporting a modest contribution from information-directed challenger selection. Along the history progression NH→NBT→Full, neural history adds 0.010 on both LongLaMP tasks, and the BT warm start adds a further 0.009 and 0.007; PrefEval remains unchanged. Thus, both history mechanisms contribute on the sequential LongLaMP tasks. The wider gaps from the random-acquisition controls reflect champion-centered collection and learned final selection.

**Persona-Construction and Density Analysis.** Keyword-conditioned construction produces denser persona spaces and lower challenger exploration for POHF and PDO Sampling across the four evaluated datasets, while ID-TAP's exploration rate changes by at most 1.5 percentage points. Appendix C reports the construction, exploration, performance, and sensitivity results.

**Robustness to Preference and Evaluation Models.** ID-TAP is numerically highest across the reported matched-model, cross-model, and BERTScore configurations, providing complementary evidence across automated preference sources, evaluators, and metrics. Appendix D provides the complete tables and protocol.

**Runtime.** Here, $K$ denotes the number of candidate personas in each user's persona space. With $K = 100$ and $n_{\text{user}} = 40$ sampled users per dataset, ID-TAP's algorithmic update requires 0.065–0.571 seconds per round, compared with 0.95–3.15 seconds of end-to-end wall-clock latency. The algorithmic component accounts for approximately 7–22% of total latency. Full timing statistics and the measurement protocol are provided in Appendix E.

## 5 Related Work

**Personalized LLM Agents.** Personalized LLMs condition outputs on individual preferences, traits, or context (Li et al., 2024b; Zhu et al., 2024). Retrieval-based methods expose relevant user history at inference time (Richardson et al., 2023; Zhong et al., 2024), while memory and prompt-based approaches maintain or synthesize user-specific context (Zhang et al., 2025a; Liu et al., 2024). LaMP standardizes evaluation across personalized language tasks (Salemi et al., 2024). Recent work increasingly studies lightweight adaptation at inference time (Li et al., 2024b; Qu et al., 2025); ID-TAP contributes a pairwise-feedback method for selecting among fixed persona prompts.

**Feedback Mechanisms and Test-Time Adaptation.** Aligning LLM behaviors with user intent relies heavily on feedback-driven optimization. The canonical paradigm, Reinforcement Learning from Human Feedback (RLHF) (Christiano et al., 2017; Ziegler et al., 2019), typically employs supervised fine-tuning, reward modeling (Stiennon et al., 2020), and policy optimization via PPO (Schulman et al., 2017; Ouyang et al., 2022). Direct Preference Optimization (DPO) (Rafailov et al., 2023) and RLAIF (Bai et al., 2022) streamline this process through direct preference learning or AI supervision. Test-Time Adaptation (TTA) further enables online refinement through iterative feedback loops (Madaan et al., 2023; Shinn et al., 2023); reasoning systems use self-correction or tree search (Yao et al., 2023; Ouyang et al., 2025), while personalized prompt- and decoding-level methods adapt behavior with fixed backbone LLM weights (Lin et al., 2024; Qu et al., 2025).

**Exploration in Dueling Bandits.** Dueling bandits learn from pairwise preferences (Yue et al., 2012), and contextual variants allow preferences to depend on the query (Dudík et al., 2015; Bengs et al., 2022). Relevant analyses cover structured stochastic preferences and neural utility models (Bengs et al., 2022; Verma et al., 2025); confidence-based linear bandit methods provide additional background for contextual exploration (Abbasi-Yadkori et al., 2011). Practical strategies include Thompson sampling (Daniel et al., 2018), Double Thompson Sampling (D-TS) (Wu & Liu, 2016), and posterior sampling for LLM exploration (Dwaracherla et al., 2024). For LLM prompt optimization, POHF (Lin et al., 2024) uses neural UCB-style acquisition, and PDO (Wu et al., 2025) combines D-TS with top-performer-guided mutation and pruning. Our PDO Sampling baseline instantiates D-TS in the common fixed persona pool and feedback budget. T-POP (Qu et al., 2025) operates during token-level decoding with white-box model access and addresses a complementary setting.

IDS balances estimated utility gaps against information gain (Russo & Van Roy, 2018) and has been studied in linear partial monitoring and dueling-feedback settings (Kirschner et al., 2020; Kirschner & Krause, 2021; Pásztor et al., 2024). These works provide general information-directed objectives or analyses under structured models. ID-TAP builds on these foundations and focuses on the systems question posed by black-box LLM personalization: scoring a large, LLM-generated persona pool, separating query-conditioned neural utility learning from lightweight hypothetical BT updates, and transferring comparison evidence across related queries. The resulting contribution is a scalable, cross-query acquisition architecture for persona-level test-time adaptation.

## 6 Conclusion

We presented **ID-TAP**, an IDS-inspired framework for test-time persona selection. It combines query-conditioned neural utility gaps, a lightweight BT entropy-reduction proxy, and similarity-weighted warm starts across related queries. Across seven user-conditioned tasks with automated proxy feedback, ID-TAP achieves the highest aggregate mean among the evaluated baselines under a fixed comparison budget. Future work will evaluate the interface with end users and develop finite-sample analysis of the neural/BT procedure.

### AI USE STATEMENT

In preparing this manuscript, generative AI tools were used to assist with LaTeX formatting, scientific-figure design and implementation, and editorial review. These tools did not generate or alter the reported experimental measurements. The authors reviewed all AI-assisted changes and take responsibility for the final text, claims, and artifacts.

### REPRODUCIBILITY STATEMENT

The main paper specifies the problem formulation, ID-TAP procedure, and evaluation protocol. The appendix provides the BT-MLE entropy-proxy implementation, complete prompts, data splits, hyperparameters, persona-space construction analyses, robustness experiments, and timing protocol needed to interpret and reproduce the reported results.

## REFERENCES

Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. *Advances in neural information processing systems*, 24, 2011.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.

Viktor Bengs, Aadirupa Saha, and Eyke Hüllermeier. Stochastic contextual dueling bandits under linear stochastic transitivity models. In *International Conference on Machine Learning*, pp. 1764–1786. PMLR, 2022.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

J Russo Daniel, Van Roy Benjamin, Kazerouni Abbas, Osband Ian, and Wen Zheng. A tutorial on thompson sampling. *Foundations and trends® in machine learning*, 11(1):1–99, 2018.

DeepSeek-AI. DeepSeek-V3.2: Pushing the frontier of open large language models. *arXiv preprint arXiv:2512.02556*, 2025. URL https://arxiv.org/abs/2512.02556.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. Enhancing chat language models by scaling high-quality instructional conversations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 3029–3051, 2023.

Miroslav Dudík, Katja Hofmann, Robert E Schapire, Aleksandrs Slivkins, and Masrour Zoghi. Contextual dueling bandits. In *Conference on Learning Theory*, pp. 563–587. PMLR, 2015.

Vikranth Dwaracherla, Seyed Mohammad Asghari, Botao Hao, and Benjamin Van Roy. Efficient exploration for llms, 2024. URL https://arxiv.org/abs/2402.00396.

Dong Huang, Jie M Zhang, Michael Luck, Qingwen Bu, Yuhao Qing, and Heming Cui. Agentcoder: Multi-agent-based code generation with iterative testing and optimisation. *arXiv preprint arXiv:2312.13010*, 2023.

David R Hunter. Mm algorithms for generalized bradley-terry models. *The annals of statistics*, 32 (1):384–406, 2004.

Johannes Kirschner and Andreas Krause. Bias-robust bayesian optimization via dueling bandits. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*. PMLR, 2021.

Johannes Kirschner, Tor Lattimore, and Andreas Krause. Information directed sampling for linear partial monitoring. In *Proceedings of the Thirty Third Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*. PMLR, 2020.

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.

Junkai Li, Yunghwei Lai, Weitao Li, Jingyi Ren, Meng Zhang, Xinhui Kang, Siyu Wang, Peng Li, Ya-Qin Zhang, Weizhi Ma, et al. Agent hospital: A simulacrum of hospital with evolvable medical agents. *arXiv preprint arXiv:2405.02957*, 2024a.

Yuanchun Li, Hao Wen, Weijun Wang, Xiangyu Li, Yizhen Yuan, Guohong Liu, Jiacheng Liu, Wenxing Xu, Xiang Wang, Yi Sun, et al. Personal llm agents: Insights and survey about the capability, efficiency and security. *arXiv preprint arXiv:2401.05459*, 2024b.

Xiaoqiang Lin, Zhongxiang Dai, Arun Verma, See-Kiong Ng, Patrick Jaillet, and Bryan Kian Hsiang Low. Prompt optimization with human feedback. *arXiv preprint arXiv:2405.17346*, 2024.

Huan Liu, Julia Qi, Zhenhao Li, Mohammad Hassanpour, Yang Wang, Konstantinos N Plataniotis, and Yuanhao Yu. Test-time personalization with meta prompt for gaze estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 3621–3629, 2024.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. Self-refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems*, 36:46534–46594, 2023.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35: 27730–27744, 2022.

Siru Ouyang, Jun Yan, I Hsu, Yanfei Chen, Ke Jiang, Zifeng Wang, Rujun Han, Long T Le, Samira Daruki, Xiangru Tang, et al. Reasoningbank: Scaling agent self-evolving with reasoning memory. *arXiv preprint arXiv:2509.25140*, 2025.

Bálint Pásztor, Parnian Kassraie, and Andreas Krause. Bandits with preference feedback: A stackelberg game perspective. In *Advances in Neural Information Processing Systems*, volume 37, pp. 11997–12034, 2024.

Zikun Qu, Min Zhang, Mingze Kong, Xiang Li, Zhiwei Shang, Zhiyong Wang, Yikun Ban, Shuang Qiu, Yao Shu, and Zhongxiang Dai. T-pop: Test-time personalization with online preference feedback. *arXiv preprint arXiv:2509.24696*, 2025.

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.

Chris Richardson, Yao Zhang, Kellen Gillespie, Sudipta Kar, Arshdeep Singh, Zeynab Raeesy, Omar Zia Khan, and Abhinav Sethy. Integrating summarization and retrieval for enhanced personalization via large language models. *arXiv preprint arXiv:2310.20081*, 2023.

Daniel Russo and Benjamin Van Roy. Learning to optimize via information-directed sampling. *Operations Research*, 66(1):230–252, 2018.

Alireza Salemi, Sheshera Mysore, Michael Bendersky, and Hamed Zamani. Lamp: When large language models meet personalization. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7370–7392, 2024.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Nihar B. Shah, Sivaraman Balakrishnan, Joseph Bradley, Abhay Parekh, Kannan Ramchandran, and Martin J. Wainwright. Estimation from pairwise comparisons: Sharp minimax bounds with topology dependence, 2015. URL https://arxiv.org/abs/1505.01462.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36:8634–8652, 2023.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. *Advances in neural information processing systems*, 33:3008–3021, 2020.

Aad W Van der Vaart. *Asymptotic statistics*, volume 3. Cambridge university press, 2000.

Arun Verma, Zhongxiang Dai, Xiaoqiang Lin, Patrick Jaillet, and Bryan Kian Hsiang Low. Neural dueling bandits: Preference-based optimization with human feedback, 2025. URL https://arxiv.org/abs/2407.17112.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Huasen Wu and Xin Liu. Double thompson sampling for dueling bandits. *Advances in neural information processing systems*, 29, 2016.

Yuanchen Wu, Saurabh Verma, Justin Lee, Fangzhou Xiong, Poppy Zhang, Amel Awadelkarim, Xu Chen, Yubai Yuan, and Shawndra Hill. Llm prompt duel optimizer: Efficient label-free prompt optimization. *arXiv preprint arXiv:2510.13907*, 2025.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. *Advances in neural information processing systems*, 36:11809–11822, 2023.

Yisong Yue, Josef Broder, Robert Kleinberg, and Thorsten Joachims. The k-armed dueling bandits problem. *Journal of Computer and System Sciences*, 78(5):1538–1556, 2012.

Weizhi Zhang, Xinyang Zhang, Chenwei Zhang, Liangwei Yang, Jingbo Shang, Zhepei Wei, Henry Peng Zou, Zijie Huang, Zhengyang Wang, Yifan Gao, et al. Personaagent: When large language model agents meet personalization at test time. *arXiv preprint arXiv:2506.06254*, 2025a.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 Embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*, 2025b. URL https://arxiv.org/abs/2506.05176.

Siyan Zhao, Mingyi Hong, Yang Liu, Devamanyu Hazarika, and Kaixiang Lin. Do llms recognize your preferences? evaluating personalized preference following in llms. *arXiv preprint arXiv:2502.09597*, 2025.

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.

Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. Large language models are human-level prompt engineers. In *The eleventh international conference on learning representations*, 2022.

Minjun Zhu, Yixuan Weng, Linyi Yang, and Yue Zhang. Personality alignment of large language models. *arXiv preprint arXiv:2408.11779*, 2024.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*, 2019.

# A Implementation Details

## A.1 Cross-Query BT Warm Start

Algorithm 2 gives the complete warm-start procedure corresponding to the normalized transfer rule in Section 3. Its history contains query-local count increments, and the normalized weights provide at most one query-equivalent of inherited mass.

**Algorithm 2** Cross-Query BT Probability Inheritance

**Input:** Current embedding $\mathbf{e}_q$; local history $\mathcal{H} = \{(q_k, \mathbf{e}_{q_k}, \Delta\mathbf{W}^{(k)}, \Delta\mathbf{N}^{(k)})\}_{k<q}$
**Output:** Initialized $(\mathbf{W}^q, \mathbf{N}^q, \mathbf{P}^q)$
**if** $\mathcal{H} = \emptyset$ **then**
  **return** $\mathbf{W}^q, \mathbf{N}^q \leftarrow \mathbf{0}$, $\mathbf{P}^q \leftarrow \mathbf{0.5}$
**end if**
$b_k \leftarrow \sigma(\beta_{\text{sim}}(\text{CosineSim}(\mathbf{e}_q, \mathbf{e}_{q_k}) - \mu_{\text{sim}}))$ for every $k < q$
$Z_q \leftarrow \max\{1, \sum_{k<q} b_k\}$; $\mathbf{W}^q, \mathbf{N}^q \leftarrow \mathbf{0}$
**for** each $(q_k, \mathbf{e}_{q_k}, \Delta\mathbf{W}^{(k)}, \Delta\mathbf{N}^{(k)}) \in \mathcal{H}$ **do**
  $a_k \leftarrow b_k / Z_q$
  $\mathbf{W}^q \leftarrow \mathbf{W}^q + a_k \Delta\mathbf{W}^{(k)}, \quad \mathbf{N}^q \leftarrow \mathbf{N}^q + a_k \Delta\mathbf{N}^{(k)}$
**end for**
$\mathbf{P}^q_{ij} \leftarrow \frac{\mathbf{W}^q_{ij} + \alpha}{\mathbf{N}^q_{ij} + 2\alpha}$
**return** $(\mathbf{W}^q, \mathbf{N}^q, \mathbf{P}^q)$

## A.2 Maintained Statistics and Smoothed Pairwise Probabilities

For each persona pair $(p_i, p_j)$, we maintain (i) directed win counts $w_{ij}$ and (ii) symmetric comparison counts $n_{ij} = n_{ji}$. After each observed duel, counts are updated as:

$$w_{ij} \leftarrow w_{ij} + \mathbb{I}\{p_i \succ p_j\}, \quad w_{ji} \leftarrow w_{ji} + \mathbb{I}\{p_j \succ p_i\}, \quad n_{ij} \leftarrow n_{ij} + 1,\ n_{ji} \leftarrow n_{ji} + 1. \tag{7}$$

Observed pairs receive the symmetric Beta-smoothed estimate

$$P_{ij} = \frac{w_{ij} + \alpha}{w_{ij} + w_{ji} + 2\alpha}, \tag{8}$$

where $\alpha = 0.7$. Unobserved pairs use $P_{ij} = 0.5$, preserving neutral pairwise evidence until a direct or inherited comparison becomes available.

## A.3 BT-MLE Strength Estimation via MM Iterations

Given the current count matrices, we estimate BT strengths $\hat{\theta}$ by an MM-style iterative solver over positive parameters $\pi_i = \exp(\theta_i)$. Let

$$W_i = \sum_{j \neq i} \tilde{w}_{ij}, \quad \tilde{w}_{ij} = w_{ij} + \alpha\, \mathbb{I}\{n_{ij} > 0\}, \tag{9}$$

and

$$\tilde{n}_{ij} = n_{ij} + 2\alpha\, \mathbb{I}\{n_{ij} > 0\}. \tag{10}$$

Each MM iteration updates

$$\pi_i \leftarrow \frac{W_i}{\sum_{j \neq i} \frac{\tilde{n}_{ij}}{\pi_i + \pi_j}}, \tag{11}$$

for candidates with positive incident comparison mass. Candidates awaiting their first observed or inherited comparison retain the common neutral initialization. The updated strengths are then normalized (sum rescaling) and converted as $\hat{\theta}_i = \log(\pi_i + \epsilon)$. For numerical stability we use $\epsilon = 10^{-8}$ and clip probabilities to $[\epsilon, 1 - \epsilon]$.

### A.4 Normalized Strength Distribution and Entropy

From $\hat{\theta}$, we define the normalized plug-in score distribution $\tilde{\pi}_t(k) = \mathrm{softmax}(\hat{\theta}_t/\tau)_k$ for the fixed query $q$ and use it to rank hypothetical updates. Unobserved pairs enter the solver with neutral pairwise evidence. Candidates awaiting their first comparison receive neutral initial strengths, and one softmax over all $K$ strengths ensures $\sum_k \tilde{\pi}_t(k) = 1$. We then compute

$$H(\tilde{\pi}_t) = -\sum_{k=1}^{K} \tilde{\pi}_t(k) \log \tilde{\pi}_t(k), \tag{12}$$

where we use the natural logarithm throughout.

### A.5 Outcome Simulation and Entropy-Reduction Proxy

For a champion–challenger pair $(c, d)$, outcome probability under BT is

$$P(y = 1 \mid c, d) = \begin{cases} \sigma(\hat{\theta}_c - \hat{\theta}_d), & n_{cd} > 0, \\ 0.5, & \text{otherwise}, \end{cases} \tag{13}$$

where $y = 1$ means champion wins.

To evaluate the proxy, we simulate both hypothetical outcomes $y \in \{0, 1\}$:

1. update $(w, n)$ with the hypothetical result;
2. re-estimate BT strengths by Eq. (11);
3. recompute the normalized strength distribution $\tilde{\pi}_t^{(y)}$ and its entropy $H(\tilde{\pi}_t^{(y)})$.

Thus, the expected post-update entropy is

$$\begin{aligned} \mathbb{E}[H_{\text{after}}] = P(y = 1) H(\tilde{\pi}_t^{(1)}) \\ + (1 - P(y = 1)) H(\tilde{\pi}_t^{(0)}), \end{aligned} \tag{14}$$

and $\tilde{g}_t(c, d) = \max\{0, H(\tilde{\pi}_t) - \mathbb{E}[H_{\text{after}}]\}$ is the plug-in entropy-reduction proxy used by the acquisition rule. Clipping at zero removes tiny negative values caused by floating-point error.

### A.6 Reproducible Procedure (Pseudo-code)

**Algorithm 3** BT-MLE Entropy-Reduction Proxy for a Candidate Pair

**Input:** champion $c$, challenger $d$, counts $(w, n)$, smoothing $\alpha$
Estimate $\hat{\theta}$ from $(w, n)$ via MM updates (Eq. 11)
Compute $\tilde{\pi}_t = \mathrm{softmax}(\hat{\theta}/\tau)$ and $H_{\text{before}} = H(\tilde{\pi}_t)$
Compute $p_{\text{win}} = P(y = 1 \mid c, d)$ using Eq. (13)
**for** $y \in \{1, 0\}$ **do**
    Simulate outcome $y$ and obtain hypothetical counts $(w^{(y)}, n^{(y)})$
    Re-estimate $\hat{\theta}^{(y)}$ from $(w^{(y)}, n^{(y)})$
    Compute $\tilde{\pi}_t^{(y)}$ and entropy $H_y = H(\tilde{\pi}_t^{(y)})$
**end for**
$\mathbb{E}[H_{\text{after}}] \leftarrow p_{\text{win}} H_1 + (1 - p_{\text{win}}) H_0$
**return** $\tilde{g}_t(c, d) = \max\{0, H_{\text{before}} - \mathbb{E}[H_{\text{after}}]\}$

**Complexity.** For each candidate challenger, the proxy performs two hypothetical BT re-estimations, one for each outcome. With a fixed number of MM iterations, an acquisition round therefore requires $2(K - 1)$ BT re-estimations.

# B EXPERIMENT DETAILS

## B.1 DATASET

**LaMP** LaMP evaluates personalization from limited user history across several language tasks. We use its two generative tasks:

- **LaMP-4 (Personalized News Headline Generation):** Given an article, generate a headline consistent with the user's historical headline style.
- **LaMP-5 (Personalized Scholarly Title Generation):** Given an abstract, generate a paper title consistent with the user's publication history.

**LongLaMP** LongLaMP extends personalized evaluation to longer-form generation. We use three tasks:

- **LongLaMP-2 (Personalized Abstract Generation):** Generate an abstract from a title and keywords in the user's academic style.
- **LongLaMP-3 (Personalized Product Review Generation):** Generate a review from a product description, rating, and review summary in the user's style.
- **LongLaMP-4 (Personalized Topic Generation):** Expand a topic summary into a Reddit post consistent with the user's history.

**PrefEval** PrefEval is a multi-turn dialogue dataset designed for evaluating preference following. We adapt it by treating the dialogue history as persona context $\mathcal{P}(\mathcal{C})$. The final user query is used as the input $q$, and the corresponding assistant response serves as the held-out reference. The agent is expected to answer $q$ while remaining consistent with the inferred user persona.

**UltraChat** UltraChat contains multi-turn human–AI conversations. We condition on the dialogue history and predict the **user's next query** $Q$, treating next-turn prediction as a user-conditioned generation task distinct from assistant-response generation.

## B.2 PROMPT TEMPLATES

This section presents all prompt templates used in ID-TAP, including persona construction, response generation, and evaluation.

**Persona Summarization Prompt.** The persona space $\mathcal{P}(\mathcal{C})$ is constructed offline from the user's historical context $\mathcal{C}$. We first generate a base persona summary using the following prompt:

**Persona Summarization Prompt**

```
=== SYSTEM ===
You are an advanced profile analysis and summarization model. Your goal is to generate
     a coherent, SPECIFIC, and psychologically insightful user persona summary.

=== USER ===
Based on the user's historical data provided below, generate an informative and
     coherent user persona summary.

### Task Instruction: {instruction}

### User History Data: {history_context}

### Instructions:
1. CORE ANCHORING: Identify 2-3 SPECIFIC, CONCRETE details from the history
2. BEHAVIORAL PATTERNS: Focus on personality traits, interests, tone, communication
     style
3. SPECIFICITY OVER ABSTRACTION: Use CONCRETE language, avoid vague terms

Please provide the final user persona summary (one paragraph, 150-300 words).
```

**Persona Rephrasing Prompt.** To construct a diverse persona space of size $K$, we iteratively rephrase the base summary. Each variant $p_i$ is generated by rephrasing the previous variant $p_{i-1}$ with bounded variation. To prevent semantic drift, we periodically reset to the original history context every $R = 15$ iterations.

**Persona Rephrasing Prompt**

```
=== SYSTEM ===
You are an advanced language model specialized in balanced rephrasing of personal
     profiles. Your task is to rewrite a user persona summary with a SLIGHTLY DIFFERENT
      perspective or emphasis.

=== USER ===
Rewrite the following user persona summary with a SLIGHTLY DIFFERENT perspective:

{previous_persona}

### STRICT Constraints:
- PRESERVE CORE ANCHORS: Keep 2-3 specific concrete details from the original
- BOUNDED VARIATION: Change emphasis or perspective, but the person should still be
     recognizable
- NO EXTREME SHIFTS: Do NOT flip personality traits dramatically
```

**Response Generation Prompt.** The following prompt template is used to generate persona-conditioned responses:

**Response Generation Prompt**

```
### Task Instruction: {instruction}

### Current Task Details: {query}

### Personality and Style Description: {persona_summary}

### Important Words from history: {keywords}

### History context: {history_entries}

### Output Requirement:
Complete the task according to the Task Instruction and Current Task Details.
     Prioritize task accuracy and relevance while using the Personality and Style
     Description to guide tone. Provide only the required response below:
```

**LLM-as-Judge Evaluation Prompt.** For pairwise preference evaluation, we employ an LLM-as-Judge approach. During online adaptation, the pair contains the two persona-conditioned responses selected for that round. For final W.R. evaluation, the pair contains the method's final response and the naive response generated directly from the query and user history or context, without persona selection or test-time preference optimization. Before each judge call, the two responses are randomly assigned to the Response 1 and Response 2 positions; the returned label is then mapped back to the corresponding source.

**LLM-as-Judge System Role**

```
You are an expert evaluator comparing two candidate responses for the same personalized
      task. The user-specific reference exemplifies the intended content and
     realization. Decide which response better fulfills the intended output as a whole.
      Jointly consider task fulfillment, semantic fidelity, organization and coherence,
      and the tone and writing style expressed by the reference, treating surface
     phrase overlap as secondary to the overall realization. Respond with ONLY a single
      digit: 1 if Response 1 is better, or 0 if Response 2 is better.
```

**Comparison Prompt**

```
Current Query: {query}

User-Specific Reference: {ground_truth}

Response 1: {response_1}

Response 2: {response_2}

Which response better realizes the intended output overall? Reply with only 1 (Response
     1 is better) or 0 (Response 2 is better). If they are equivalent overall, return
    0 by default. Please provide the final result (1 or 0 only) below:
```

### B.3 Data Splitting for LaMP Datasets

For LaMP-based datasets (LaMP-4, LaMP-5, LongLaMP-2, LongLaMP-3, LongLaMP-4), each user's historical profile is split into two parts: (1) **history context** used for persona generation, and (2) **input/output pairs** used as test queries with held-out reference responses.

**Splitting Rules.** Let $n$ denote the number of profile entries for a user. The splitting follows a threshold-based strategy:

- If $n < T_{\text{small}}$: The first $H_{\text{small}}$ entries form the history context; the remaining entries become input/output pairs.
- If $T_{\text{small}} \leq n < T_{\text{large}}$: The first $H_{\text{large}}$ entries form the history context; the remaining entries become input/output pairs.
- If $n \geq T_{\text{large}}$: The first $H_{\text{large}}$ entries form the history context; entries from position $H_{\text{large}}+1$ to $H_{\text{large}} + (N_{\text{IO}} - 1)$ become input/output pairs (capped at $N_{\text{IO}} - 1$ additional pairs).

Table 3: Data Splitting Parameters

| Parameter | Symbol | Value |
|---|---|---|
| Small profile threshold | $T_{\text{small}}$ | 10 |
| Large profile threshold | $T_{\text{large}}$ | 20 |
| History context count (small) | $H_{\text{small}}$ | 5 |
| History context count (large) | $H_{\text{large}}$ | 10 |
| Total I/O pairs limit | $N_{\text{IO}}$ | 10 |

**Query and Output Extraction.** For each dataset, queries and held-out reference outputs are extracted from profile entries as follows:

- **LaMP-4** (News Headline): Query = article text; Output = headline title
- **LaMP-5** (Paper Title): Query = abstract; Output = paper title
- **LongLaMP-2** (Abstract Generation): Query = paper title; Output = abstract
- **LongLaMP-3** (Product Review): Query = product description + rating; Output = review text
- **LongLaMP-4** (Reddit Post): Query = topic summary; Output = post content

The history context supports persona summarization and construction of $\mathcal{P}(\mathcal{C})$. Each input defines a test query, while its held-out output conditions the offline proxy user and evaluation. During adaptation, the proxy user retains this reference and ID-TAP receives the resulting binary comparison.

Table 4: Unnormalized ROUGE-L performance before per-query normalization. Results are mean±SEM across users, where each user's result averages three independent runs. **Bold** indicates the best result and underlined values indicate the second-best result in each dataset.

| Dataset | $N_{\text{eval}}$ | PersonaAgent | POHF | PDO Samp. | ID-TAP |
|---|---|---|---|---|---|
| LaMP-4 | 363 | 0.1700±0.007846 | 0.1790±0.007499 | 0.1768±0.007085 | **0.1973±0.006559** |
| LaMP-5 | 400 | 0.3924±0.009460 | 0.4385±0.008671 | 0.4346±0.008551 | **0.4927±0.008513** |
| LongLaMP-2 | 400 | 0.1729±0.002141 | 0.1716±0.002119 | 0.1678±0.001918 | **0.1831±0.002042** |
| LongLaMP-3 | 230 | 0.1322±0.001532 | 0.1391±0.001712 | 0.1378±0.001251 | **0.1456±0.001562** |
| LongLaMP-4 | 326 | 0.1217±0.001576 | 0.1234±0.001806 | 0.1234±0.001547 | **0.1367±0.001853** |
| PrefEval | 40 | 0.1700±0.005887 | 0.2103±0.006989 | 0.1943±0.004623 | **0.2131±0.007093** |
| UltraChat | 40 | 0.0758±0.005492 | 0.1859±0.009107 | 0.1815±0.009301 | **0.1909±0.008171** |
| **Average** | – | 0.1764 | 0.2068 | 0.2023 | **0.2228** |

### B.4 Unnormalized ROUGE-L Statistics

For each user, raw (unnormalized) ROUGE-L is averaged over three independent runs. Table 4 reports the mean and SEM across these per-user results. $N_{\text{eval}}$ denotes the number of query-level evaluation instances contributing to the user-level aggregates. In our logging notation, *lamp8/lamp9/lamp10* correspond to *LongLaMP-2/LongLaMP-3/LongLaMP-4*.

ID-TAP has the highest raw ROUGE-L mean on all seven datasets, with a cross-dataset average of 0.2228 compared with 0.2068 for the strongest baseline. These unnormalized values provide a scale-preserving counterpart to the normalized main results.

### B.5 Persona Space Quality Analysis

To characterize the generated persona spaces, Table 5 reports five geometric metrics for all seven benchmarks. The analysis covers 180 users, with 400 candidate personas per user. Here, $N_{\text{user}}$ is 20 for each LaMP/LongLaMP dataset and 40 for PrefEval and UltraChat.

**Density & Spread.** The mean pairwise cosine similarity $\bar{\mu}$ ranges from 0.657 (LongLaMP-3) to 0.716 (PrefEval), with standard deviations $\bar{\sigma}$ between 0.080 and 0.093. The 10th–90th percentile interval ($p_{10}$–$p_{90}$) spans roughly 0.22 in each dataset (e.g., [0.537, 0.769] for LongLaMP-3). These statistics describe moderately dense but non-saturated embedding spaces.

**Redundancy.** The near-duplicate rate (fraction of pairs with cosine similarity $\geq 0.95$) stays below 0.20%, and the effective-arm ratio—the percentage of candidates whose nearest-neighbor similarity is below 0.95—ranges from 94.4% to 98.2%. Under this threshold, the pools contain little near-duplicate redundancy while retaining many distinct candidate embeddings.

**Geometric Structure.** K-means silhouette scores at $K=20$ range from 0.097 (UltraChat) to 0.125 (LaMP-5), indicating weak separation into discrete clusters. Together with the similarity statistics, these scores characterize the candidate pools as dense and weakly clustered; the acquisition experiments measure the corresponding method-level behavior.

### B.6 Hyperparameters

This section details all hyperparameters used in ID-TAP, organized into four categories: neural network architecture, training configuration, algorithm-specific parameters, and LLM settings. Table 6 provides a comprehensive summary.

**Neural Network Architecture.** The utility function $f_\theta(q, p)$ is parameterized by a single-hidden-layer feedforward neural network. The input consists of the concatenation of query embedding $\phi(q) \in \mathbb{R}^{d_{\text{emb}}}$ and persona embedding $\phi(p) \in \mathbb{R}^{d_{\text{emb}}}$, where $d_{\text{emb}} = 1024$ for standard mode and $d_{\text{emb}} = 2048$ for contextual mode. We employ GELU activation for its smooth gradient properties

Table 5: Persona space quality metrics across 7 benchmarks (180 users in total; 400 candidates per user). $N_{\text{user}}$ is the number of sampled users in each dataset. $\bar{\mu}$: mean pairwise cosine similarity; $\bar{\sigma}$: standard deviation; $p_{10}$/$p_{90}$: 10th/90th percentile of the pairwise similarity distribution; Near-dup: fraction of pairs with similarity $\geq 0.95$; Eff. Arm: percentage of candidates whose nearest-neighbor similarity $< 0.95$; Silhouette: K-means silhouette coefficient at $K$=20.

| Dataset | $N_{\text{user}}$ | $\bar{\mu}$ | $\bar{\sigma}$ | $p_{10}$ | $p_{90}$ | Near-dup $\geq$0.95 (%) | Eff. Arm (%) | Silhouette ($K$=20) |
|---|---|---|---|---|---|---|---|---|
| LaMP-4 | 20 | 0.672 | 0.090 | 0.553 | 0.779 | 0.041 | 97.6 | 0.104 |
| LaMP-5 | 20 | 0.694 | 0.081 | 0.590 | 0.790 | 0.094 | 95.3 | 0.125 |
| LongLaMP-2 | 20 | 0.701 | 0.080 | 0.599 | 0.794 | 0.092 | 95.0 | 0.121 |
| LongLaMP-3 | 20 | 0.657 | 0.093 | 0.537 | 0.769 | 0.018 | 98.2 | 0.107 |
| LongLaMP-4 | 20 | 0.686 | 0.084 | 0.577 | 0.787 | 0.037 | 97.5 | 0.105 |
| PrefEval | 40 | 0.716 | 0.083 | 0.607 | 0.814 | 0.190 | 94.4 | 0.098 |
| UltraChat | 40 | 0.715 | 0.087 | 0.600 | 0.819 | 0.158 | 94.5 | 0.097 |

and use a dropout rate of $0.05$ for regularization. Weights are initialized using Kaiming normal initialization to ensure proper gradient flow at initialization.

**Training Configuration.** The neural network is trained using the AdamW optimizer with a learning rate of $5 \times 10^{-4}$ and weight decay of $10^{-3}$. We use a cosine annealing learning rate scheduler that decays the learning rate to 50% of its initial value over the training epochs. Each update consists of 100 epochs with a batch size of 4, and gradient clipping with a maximum norm of 1.0 is applied to prevent gradient explosion during early training stages when comparison data may be noisy.

**ID-TAP Algorithm Parameters.** The acquisition ratio in Section 3 uses a numerical floor $\epsilon_g = 10^{-8}$. The BT explorer uses smoothing $\alpha = 0.7$ and softmax temperature $\tau = 4.5$ for the normalized strength distribution. Cross-query transfer uses $\beta_{\text{sim}} = 6$ and $\mu_{\text{sim}} = 0.8$ in the normalized similarity weights.

**LLM Configuration.** The main configuration uses DeepSeek-V3.2 for response generation and for the LLM-based preference proxy and final evaluation; the ROUGE-L-based simulator follows the scoring rule in Section 4.3. Appendix D varies the preference and evaluation models. Generation uses temperature 0.6, whereas LLM-as-Judge evaluation uses temperature 0.0. The maximum output length is 1024 tokens, with frequency and presence penalties of 0.5.

Table 6: Complete hyperparameter configuration for ID-TAP. The table is organized into four sections: neural network architecture, training settings, algorithm-specific parameters, and LLM configuration.

| Category | Parameter | Symbol | Value |
|---|---|---|---|
| *Neural Network* | Input dimension | $d_{\text{in}}$ | $2 \times d_{\text{emb}}$ |
| | Hidden dimension | $d_h$ | 1024 |
| | Number of hidden layers | $L$ | 1 |
| | Activation function | $\sigma(\cdot)$ | GELU |
| | Dropout rate | $p_{\text{drop}}$ | 0.05 |
| *Training* | Learning rate | $\eta$ | $5 \times 10^{-4}$ |
| | Optimizer | – | AdamW |
| | Weight decay | $\lambda$ | $10^{-3}$ |
| | Batch size | $B$ | 4 |
| | Update epochs | – | 100 |
| | Gradient clipping | – | 1.0 |
| | Learning-rate schedule | – | Cosine to $0.5\eta$ |
| *ID-TAP Algorithm* | BT smoothing | $\alpha$ | 0.7 |
| | Acquisition floor | $\epsilon_g$ | $10^{-8}$ |
| | Softmax temperature | $\tau$ | 4.5 |
| | Sigmoid steepness | $\beta_{\text{sim}}$ | 6 |
| | Sigmoid midpoint | $\mu_{\text{sim}}$ | 0.8 |
| *LLM Settings* | Model | – | DeepSeek-V3.2 |
| | Temperature (generation) | – | 0.6 |
| | Temperature (judge) | – | 0.0 |
| | Max output tokens | – | 1024 |
| | Frequency penalty | – | 0.5 |
| | Presence penalty | – | 0.5 |

# C Persona-Construction and Density Analysis

## C.1 Construction and Exploration Diagnostic

For each dataset, we compare iterative rephrasing (IR) with keyword-conditioned construction (KW) using a common candidate budget, response generator, embedding model, interaction budget, and evaluation protocol. IR follows the persona rephrasing procedure in Section B.2, including the reset interval $R = 15$. KW extracts preference and style keywords from the same user history, assigns keyword combinations to candidate arms, and generates a persona conditioned on each combination. Both constructions use Qwen3-Embedding-4B representations before online optimization.

For method $m$, let $\mathcal{H}_{t-1}^{q,m} = \{p_{1,2}^{q,m}, \ldots, p_{t-1,2}^{q,m}\}$ denote the challengers selected before round $t$. The Arm-2 exploration rate is

$$\text{ExplorationRate}(m) = \frac{1}{T-1} \sum_{t=2}^{T} \mathbb{I}\left[p_{t,2}^{q,m} \notin \mathcal{H}_{t-1}^{q,m}\right]. \tag{15}$$

The first round is excluded because its challenger is necessarily unseen. A repeated challenger is counted as exploitation, and the same tie-handling implementation is used for all methods.

Table 7: Mean pairwise cosine similarity under iterative rephrasing (IR) and keyword-conditioned construction (KW).

| Dataset | $\bar{\mu}$ (IR) | $\bar{\mu}$ (KW) | $\Delta\bar{\mu}$ |
|---|---|---|---|
| LaMP-4 | 0.672 | 0.720 | $+0.048$ |
| LongLaMP-4 | 0.686 | 0.695 | $+0.009$ |
| PrefEval | 0.716 | 0.830 | $+0.114$ |
| UltraChat | 0.715 | 0.834 | $+0.119$ |

Table 8: Arm-2 exploration rates under iterative rephrasing (IR).

| Dataset | POHF | PDO Samp. | ID-TAP |
|---|---|---|---|
| LaMP-4 | 24.1 | 25.7 | 73.5 |
| LongLaMP-4 | 25.3 | 28.1 | 72.2 |
| PrefEval | 65.0 | 64.0 | 95.3 |
| UltraChat | 67.3 | 66.8 | 94.8 |

Table 9: Arm-2 exploration rates under keyword-conditioned construction (KW). Parentheses report the change from IR in percentage points.

| Dataset | POHF | PDO Samp. | ID-TAP |
|---|---|---|---|
| LaMP-4 | 16.6 ($-7.5$) | 17.6 ($-8.1$) | 74.8 ($+1.3$) |
| LongLaMP-4 | 16.5 ($-8.8$) | 20.3 ($-7.8$) | 71.9 ($-0.3$) |
| PrefEval | 61.0 ($-4.0$) | 62.5 ($-1.5$) | 94.4 ($-0.9$) |
| UltraChat | 64.8 ($-2.5$) | 60.5 ($-6.3$) | 93.3 ($-1.5$) |

Across all four datasets, KW increases mean persona similarity and is accompanied by lower exploration for POHF and PDO Sampling. ID-TAP's exploration rate changes by at most 1.5 percentage points between the two constructions. Together, these results show an association between denser persona spaces and reduced conventional exploration, alongside comparatively stable exploration by ID-TAP.

### C.2 Exploration–Performance Association

Table 10: Association between conventional exploration under KW and ID-TAP's ROUGE-L margin over the strongest baseline.

| Dataset group | Mean POHF exploration | Mean ID-TAP margin |
|---|---|---|
| LaMP-4 and LongLaMP-4 | 16.55% | $+0.0160$ |
| PrefEval and UltraChat | 62.90% | $+0.0065$ |

The datasets with lower POHF exploration under KW also have the larger reported ID-TAP margin. Together with the within-dataset IR–KW comparison, this descriptive pattern aligns lower conventional exploration with a larger performance margin for ID-TAP.

### C.3 Sensitivity to Persona-Space Size and Construction

In the following tables, $n_{\text{user}} = 40$ denotes sampled users per dataset. These experiments use $K = 100$ candidates, whereas the main experiments use $K = 400$.

Table 11: ROUGE-L with iterative-rephrasing personas ($K = 100$, $n_{\text{user}} = 40$ sampled users per dataset).

| Dataset | ID-TAP | POHF | PDO Samp. | PersonaAgent |
|---|---|---|---|---|
| UltraChat | **0.2430** | 0.2291 | 0.2213 | 0.1057 |
| PrefEval | **0.2587** | 0.2498 | 0.2430 | 0.1651 |
| LaMP-4 | **0.2262** | 0.1900 | 0.2001 | 0.1529 |
| LongLaMP-4 | **0.1392** | 0.1310 | 0.1320 | 0.1230 |

Table 12: ROUGE-L with keyword-conditioned personas ($K = 100$, $n_{\text{user}} = 40$ sampled users per dataset).

| Dataset | ID-TAP | POHF | PDO Samp. | PersonaAgent |
|---|---|---|---|---|
| UltraChat | **0.2565** | 0.2441 | 0.2496 | 0.1090 |
| PrefEval | **0.2614** | 0.2556 | 0.2527 | 0.1647 |
| LaMP-4 | **0.2495** | 0.2201 | 0.2172 | 0.1554 |
| LongLaMP-4 | **0.1388** | 0.1356 | 0.1350 | 0.1253 |

At $K = 100$, ID-TAP achieves the highest mean ROUGE-L under both candidate-space constructions, extending the main $K = 400$ result to a smaller persona pool and an alternative construction procedure.

## D Robustness to Preference and Evaluation Models

We evaluate three settings: GPT-4o for both preference generation and final evaluation, GPT-4o for preference generation with DeepSeek-V3.2 for final evaluation, and BERTScore as an evaluation metric independent of the optimizer's preference signal. Each setting follows the main protocol for candidate pools, interaction budgets, response generation, prompts, and aggregation, while varying the specified preference or evaluation component.

Table 13: Win rate with GPT-4o used for both preference generation and evaluation.

| Dataset | PersonaAgent | POHF | PDO Samp. | ID-TAP |
|---|---|---|---|---|
| LaMP-4 | 73.2 | 80.2 | 80.4 | **81.9** |
| UltraChat | 81.7 | 86.7 | 85.8 | **89.1** |

Table 14: Win rate with GPT-4o preferences and DeepSeek-V3.2 evaluation.

| Dataset | PersonaAgent | POHF | PDO Samp. | ID-TAP |
|---|---|---|---|---|
| LaMP-4 | 68.1 | 75.0 | 77.2 | **78.2** |
| UltraChat | 64.2 | 85.0 | 82.5 | **89.2** |

Table 15: BERTScore under a metric independent of the preference-optimization signal.

| Dataset | PersonaAgent | POHF | PDO Samp. | ID-TAP |
|---|---|---|---|---|
| UltraChat | 0.838 | 0.890 | 0.889 | **0.892** |
| PrefEval | 0.847 | 0.885 | 0.887 | **0.888** |
| LaMP-4 | 0.862 | 0.870 | 0.872 | **0.876** |
| LongLaMP-4 | 0.812 | 0.814 | 0.812 | **0.818** |

ID-TAP achieves the highest reported mean in every configuration. The matched GPT-4o setting changes the automated preference source and final evaluator together, the GPT-4o/DeepSeek-V3.2 setting separates those roles, and BERTScore evaluates responses independently of the preference-optimization signal. The resulting ordering is consistent across the three evaluation configurations.

# E Runtime and Timing Protocol

Timing measurements were collected on a workstation equipped with a single NVIDIA GeForce RTX 4090 GPU using the model-serving backend, batch behavior, and concurrency configuration of the main experiments. Here, $K$ denotes the number of candidate personas in each user's persona space, and $n_{\text{user}}$ denotes the number of sampled users per dataset. The measurements use $K = 100$ and $n_{\text{user}} = 40$. Each user's timing is averaged over three independent runs, and the tables report the mean, standard deviation, and SEM across these per-user averages. Wall-clock time covers a complete interaction round, including response generation and evaluation; update time isolates persona selection, BT-MLE estimation, neural-model updating, and entropy-proxy computation.

Table 16: ID-TAP wall-clock time with $K = 100$ and $n_{\text{user}} = 40$ sampled users per dataset. Statistics are computed across per-user averages of three independent runs and reported in seconds per interaction round.

| Dataset | Mean | Std. | SEM |
|---|---|---|---|
| UltraChat | 1.1142 | 0.4211 | 0.0666 |
| PrefEval | 0.9513 | 0.1745 | 0.0276 |
| LaMP-4 | 2.5674 | 0.9069 | 0.1434 |
| LongLaMP-4 | 3.1499 | 0.9750 | 0.1542 |

Table 17: ID-TAP algorithmic update time with $K = 100$ and $n_{\text{user}} = 40$ sampled users per dataset. Statistics are computed across per-user averages of three independent runs and reported in seconds per interaction round.

| Dataset | Mean | Std. | SEM |
|---|---|---|---|
| UltraChat | 0.0964 | 0.0189 | 0.0030 |
| PrefEval | 0.0651 | 0.0157 | 0.0025 |
| LaMP-4 | 0.5709 | 0.1459 | 0.0231 |
| LongLaMP-4 | 0.3172 | 0.0839 | 0.0133 |

Algorithmic updates require 0.065–0.571 seconds per round, compared with 0.95–3.15 seconds end to end. Based on the reported means, the algorithmic component accounts for approximately 7–22% of total latency, with model serving dominating the measured wall-clock time.

# F Persona Examples

## F.1 LaMP-4

**Persona A**

This user approaches political coverage as a strategic observer, with a particular focus on the mechanics of the 2016 Republican primary. Their analytical lens is drawn to the concrete maneuvers and positional conflicts between figures like Donald Trump, Ted Cruz, and Jeb Bush. They consistently select content detailing the "neck-and-neck battle" in Iowa or the logistical specifics of a vice-presidential search, prioritizing articles that dissect how candidates compete. Their neutral tone and preference for headlines framing events as consequential clashes—a "fiery debate" or a "growing feud"—reveal a perspective that treats politics as a complex game. The core interest remains in tracking concrete developments, such as poll results and campaign strategies that signal who might "knock out" rivals.

**Persona B**

This user approaches political campaigns as intricate systems to be reverse-engineered, with a focus on how discrete operational choices cascade into winning or losing outcomes. Their detailed study of the 2016 Republican primary, particularly the delegate calculations that defined the Trump–Cruz dynamic in Iowa, serves as a foundational case. They treat strategic events—like Marco Rubio's decision to decline a vice-presidential spot—not as political drama but as pivotal system inputs that reroute a campaign's entire pathway. Polling shifts and tactical proposals are analyzed as interconnected variables within a dynamic model.

**Persona C**

The user is a politically observant individual with a concentrated focus on the 2016 Republican presidential primary's competitive landscape. Their interest centers on key figures like Donald Trump, Ted Cruz, Jeb Bush, and Marco Rubio, with attention to pivotal moments such as Trump's vice-presidential search involving Mike Pence and the "neck-and-neck battle" in Iowa polls. They consistently favor content analyzing electoral mechanics and candidate positioning—tracking how a Trump sweep could "knock out" rivals in Florida and Ohio. Their chosen material frames events through a lens of conflict and consequence, highlighting a "growing feud," a "fiery debate," or campaigns "scrambling to make up ground."

Figure 3: Three persona descriptions sampled from the same user domain in **LaMP-4** (news headline generation), illustrating the diversity of summaries maintained in the persona bank.

### F.2 LONGLAMP-4

**Persona A**

This user is a systematic thinker who values logical consistency, clear hierarchies, and functional design across diverse domains. They critique *Minecraft*'s development for lacking direction, advocating for a structured "wood–stone–iron–diamond" material system over random additions like the "bizarre endgame" and poorly implemented potions. Their analytical approach extends to physics, explaining complex concepts like 4-vectors by simplifying them into core principles. They exhibit strong, evidence-backed opinions, defending financial markets as providing liquidity and downward price pressure, while dismissing overly broad definitions of art. Their communication is assertive and detailed, often using personal comparisons—like favoring a $700 HP laptop over a $1500 MacBook—to ground arguments in pragmatic cost-benefit analysis. This pattern reveals a persona deeply invested in deconstructing systems, prioritizing structured understanding and practical utility over abstract or sentimental reasoning.

**Persona B**

This thinker approaches systems with a forensic eye for functional hierarchy and practical coherence. Their satisfaction comes from mapping how components interlock to produce reliable outcomes, whether in game design's reward loops or technology's price-to-performance ratios. They defend *Minecraft*'s clean "wood–stone–iron–diamond" progression as logically sound architecture, critiquing later potion systems as messy deviations. Their analysis of consumer tech, such as comparing a $700 HP laptop to a $1500 MacBook, seeks tangible benchmarks over brand prestige. Communication is direct and dismantling: they prefer explaining complex phenomena—like the constancy of light via 4-vectors or financial instruments as liquidity tools—through grounded mechanics. For them, rigor is the removal of sentimental abstraction to reveal the operative skeleton beneath.

**Persona C**

This user is fundamentally a model-builder, driven to synthesize disparate domains into coherent, actionable frameworks. Their analytical skill is applied to practical puzzles, from evaluating laptop specifications to dissecting *Minecraft* enchantment systems, treating each as a case study in translating convoluted rules into functional logic. While conversant with advanced concepts like 4-vectors, their true talent lies in integration—weaving threads from art criticism and financial theory to construct robust mental models. Their confidence is pragmatic, rooted in the belief that more elegant, integrated systems are not merely ideal but essential for progress. They operate as a pragmatic architect of knowledge, specifically focused on the intersection where technological systems meet economic principles.

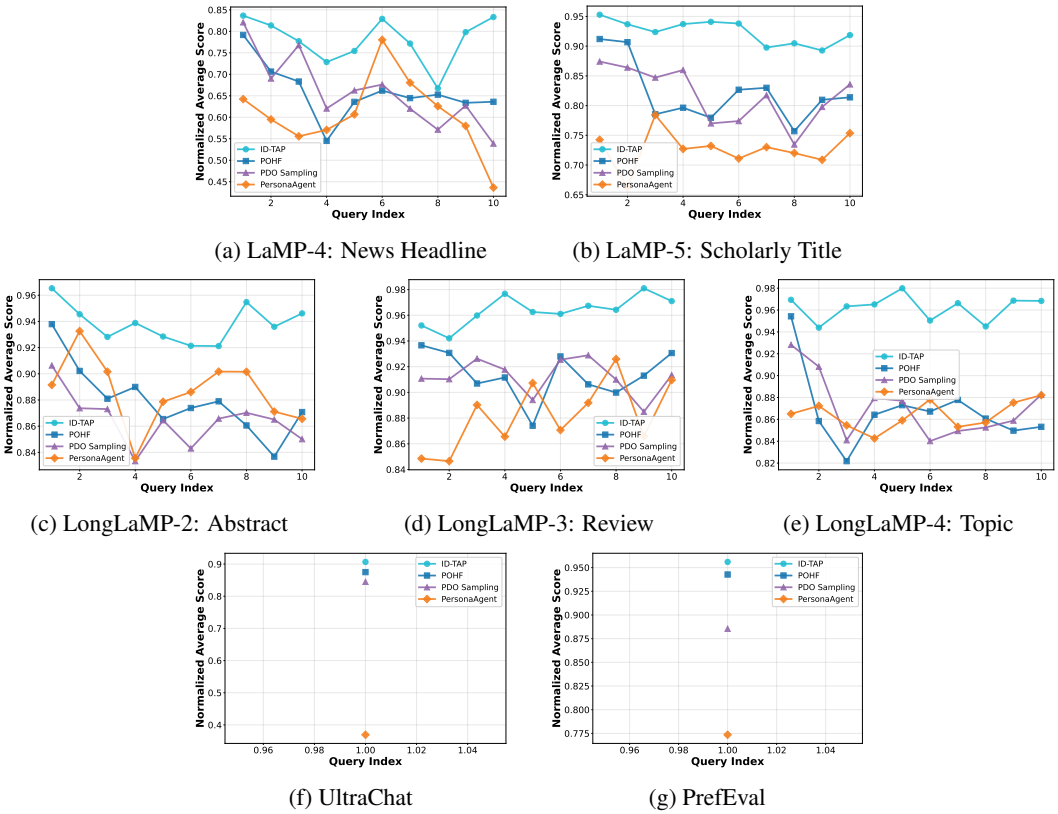
Figure 4: Three persona descriptions sampled from the same user domain in **LongLaMP-4** (topic-focused long-form writing). The user domain spans game design critique, consumer technology analysis, physics exposition, and economic commentary.

# G More Experimental Results

Figure 5 reports the individual-dataset results underlying the aggregate view in the main paper. Figure 6 shows how performance evolves within the fixed comparison budget and provides dataset-specific detail beyond the aggregate curve.

## G.1 Detailed Performance Comparison Across Datasets

Figure 5 compares ID-TAP across three benchmark categories: (1) **LaMP**, covering personalized headline and scholarly-title generation; (2) **LongLaMP**, covering personalized abstract, review, and topic generation; and (3) **multi-turn dialogue**, covering UltraChat and PrefEval.




(a) LaMP-4: News Headline (b) LaMP-5: Scholarly Title

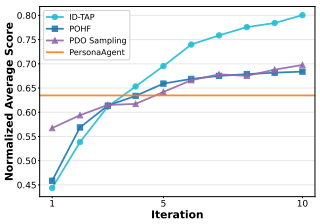
(c) LongLaMP-2: Abstract (d) LongLaMP-3: Review (e) LongLaMP-4: Topic

(f) UltraChat (g) PrefEval

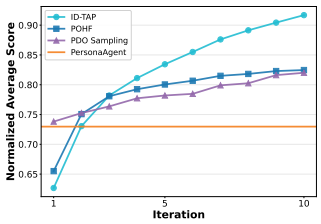
Figure 5: Query-level performance across all evaluated benchmarks.

## G.2 Detailed Iteration-Level Trajectories

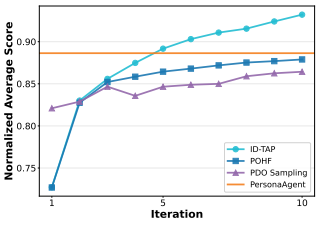
Figure 6 presents the iteration-level trajectory for each benchmark under automated proxy feedback.



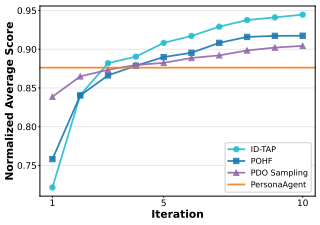

(a) LaMP-4: News Headline



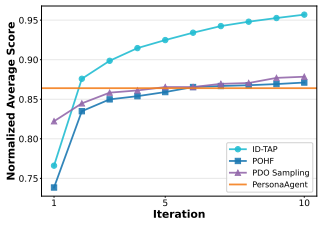

(b) LaMP-5: Scholarly Title



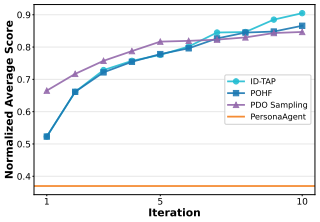

(c) LongLaMP-2: Abstract



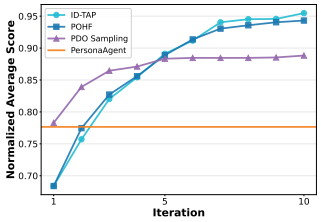

(d) LongLaMP-3: Review




(e) LongLaMP-4: Topic




(f) UltraChat




(g) PrefEval

Figure 6: Iteration-level trajectories across all evaluated benchmarks.