Instant Personalized Large Language Model Adaptation via Hypernetwork

Anonymous ACL submission

Abstract

Personalized large language models (LLMs) tailor content to individual preferences using user profiles or histories. However, existing parameter-efficient fine-tuning (PEFT) methods, such as the "One-PEFT-Per-User" (OPPU) paradigm, require training a separate adapter for each user, making them computationally expensive and impractical for real-time updates. We introduce Profile-to-PEFT, a scalable framework that employs a hypernetwork, trained end-to-end, to map a user's encoded profile directly to a full set of adapter parameters (*e.g.*, LoRA), eliminating per-user training at deployment. This design enables instant adaptation, generalization to unseen users, and privacy-preserving local deployment. Experimental results demonstrate that our method outperforms both prompt-based personalization and OPPU while using substantially fewer computational resources at deployment. The framework exhibits strong generalization to out-of-distribution users and maintains robustness across varying user activity levels and different embedding backbones. The proposed Profile-to-PEFT framework enables efficient, scalable, and adaptive LLM personalization suitable for large-scale applications.

1 Introduction

Personalization aims to tailor system interactions, content, and recommendations to a user's specific needs and preferences by leveraging their historical data (Tan and Jiang, 2023; Chen et al., 2024; Kirk et al., 2024; Liu et al., 2025). While large language models (LLMs) have demonstrated powerful generative capabilities, their general-purpose, "one-size-fits-all" nature limits their ability to cater to individual users (Guan et al., 2025; Zhang et al., 2024). Consequently, integrating the generative strength of LLMs with user-specific personalization has become a critical research direction (Li et al., 2023; Jiang et al., 2025; Tan et al., 2025a).

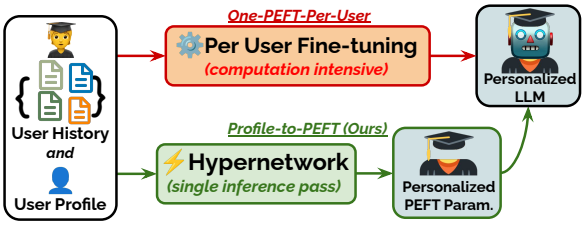

Figure 1: The "One-PEFT-Per-User" method uses computationally intensive fine-tuning to create personalized parameters. In contrast, our proposed Profile-to-PEFT uses a hypernetwork to directly generate parameters from user history or profile in a single inference pass.

Recent methodologies fall into two main categories: prompt-based and parameter-efficient fine-tuning (PEFT)-based. Prompt-based personalization techniques design specific prompt templates to guide LLMs in capturing user preferences, encompassing approaches such as vanilla personalized prompting (Zhiyuli et al., 2023), retrieval-augmented prompting (Salemi et al., 2024b), and profile-augmented prompting (Richardson et al., 2023). However, these methods expose user data to centralized LLMs, raising significant concerns regarding user privacy and hindering model ownership for deep personalization (Tan et al., 2024b). Additionally, prompt-based techniques are susceptible to distraction by irrelevant user historical data, an issue difficult to mitigate solely through incorporating additional retrieval context (Shi et al., 2023). Conversely, PEFT-based personalization strategies store users' preferences and behavioral patterns in lightweight, user-specific parameters. OPPU (Tan et al., 2024b) is a pioneering PEFT-based approach that effectively encodes user preferences into individual PEFT parameters, facilitating model ownership, stronger personalization performance, and superior generalization of user behavior patterns compared to prompt-based methods.

Despite their effectiveness, existing PEFT-based frameworks operate under a "one-PEFT-per-user" paradigm, where a unique module is trained from

scratch for each user. This approach presents substantial computational and scalability challenges, particularly in large-scale systems with millions of users or in dynamic settings where user preferences evolve continuously. Training or updating individual PEFT modules in real-time is computationally prohibitive. This bottleneck leads to a key research question: *Is it possible to generate personalized PEFT parameters directly from a user's profile in an efficient step, thereby eliminating the need for per-user training at deployment?*

To address this challenge, we introduce Profile-to-PEFT (P2P), a novel framework that learns a direct mapping from user profiles to personalized PEFT parameters. Instead of relying on iterative fine-tuning, our method utilizes a hypernetwork that generates a full set of personalized LoRA adapter weights conditioned on a user's profile, thereby eliminating the need to perform per user training at deployment, as illustrated in Figure 1. The process, detailed in Figure 2, begins by constructing the user profile composed of natural language user preference summaries from user history and retrieved historical interactions, then compact the profile into user embeddings. This embedding, augmented with learnable position and module identifiers, is then fed into an MLP-based hypernetwork, which outputs the entire set of personalized adapter parameters in a single forward pass. By plugging in the user-specific parameters and train this framework end-to-end on a diverse user population data using supervised finetuning, P2P learns to generalize across unseen users.

We conduct extensive experiments on LaMP (Salemi et al., 2024b), LongLaMP (Kumar et al., 2024), Personal Reddit (Staab et al., 2023), and Empathic Conversations (Omitaomu et al., 2022) datasets that containing diverse classification and generation tasks. The results, corroborated by LLM-as-a-Judge evaluations, demonstrate that P2P generalizes effectively to both in-distribution and out-of-distribution users. Our analyses confirm that training user diversity is more critical than sheer quantity for robust performance and that our generation-based approach is 33x faster at deployment than the OPPU paradigm. Further studies validate the framework's robustness to different embedding models and varying user activity levels, confirming our key design choices.

In summary, the proposed P2P framework advances PEFT-based personalized LLM towards practical deployment at industrial scales, enabling

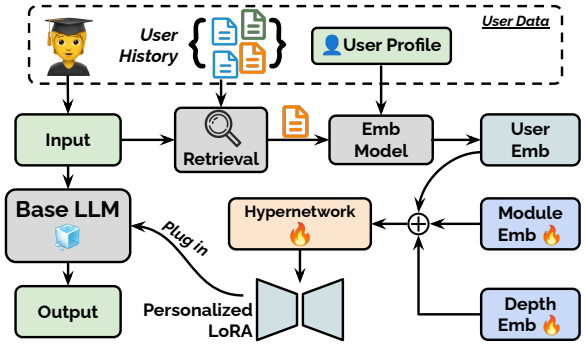

Figure 2: Overview of the Profile-to-PEFT architecture, where user history, depth, module embeddings are fed into the hypernetwork to obtain personalized LoRA. P2P is optimized in a end-to-end training manner.

strong generalization to unseen users during training, and achieve real-time personalized adaptations of LLMs. P2P maintains user privacy and significantly reduces the computational burden and carbon footprint of personalized LLM training.

2 Preliminary

Research Problem Formulation We aim to personalize LLMs for individual users. At time t, a user u with history $\mathcal{H}_u^t$ (containing all behaviors before t) queries the model with input x_u to receive personalized output y_u. The goal is to obtain personalized parameters ΔW_u for each user u or ΔW_{x_u} for each input x_u of user u.

Low-Rank Adaptation (LoRA) (Hu et al., 2021) is a PEFT method that freezes pre-trained weights of a LLM $W_0 \in \mathbb{R}^{d_{out} \times d_{in}}$ and introduce trainable low-rank matrices $\Delta W = BA$, where $B \in \mathbb{R}^{d_{out} \times r}$ and $A \in \mathbb{R}^{r \times d_{in}}$, with the rank $r \ll \min(d_{in}, d_{out})$. The model's forward pass becomes $h = W_0 x + \Delta W x = W_0 x + BAx$. We denote LoRA weights for module m at layer l as $\Delta W^{m,l}$.

Personalization via Per-User PEFT (Tan et al., 2024b) trains unique PEFT parameters for each user by using the following objective:

$$\Delta W_u^* = \arg\min_{\Delta W} \mathcal{L}_{\text{SFT}}(\Psi \oplus \Delta W, \mathcal{H}_u^{<t}),$$

where Ψ denotes frozen base model weights, $\mathcal{L}_{\text{SFT}}$ is the supervised fine-tuning loss, and $\oplus$ applies PEFT to the base model. While effective, this requires separate training for every user, limiting scalability and real-time adaptation.

3 Profile-to-PEFT (P2P)

To address the limitations of the one-PEFT-per-user paradigm, we present Profile-to-PEFT (P2P), which learns a direct mapping from a user profile to PEFT parameters. Instead of running iterative per-user optimization, P2P uses a hypernetwork f_θ to produce personalized LoRA weights in a single forward pass, as illustrated in Figure 2.

3.1 Model Architecture

P2P generates the LoRA matrices $(A_{x_u}^{m,l}, B_{x_u}^{m,l})$ for each target module m at layer l conditioned on a user profile. This process involves three key steps.

User Profile Encoding First, a textual user profile $p_u^{<t}$ is constructed to dynamically represent user preferences. This profile combines a global summary $s_u^{<t} = \text{Profiler}(\mathcal{H}_u^{<t})$, generated from the user's history by the base LLM, with the top-k most relevant historical interactions retrieved by a retriever $\mathcal{R}$ conditioned on the current input x_u. The profile text is formulated as:

$$p_{x_u}^{<t} = \left[s_u^{<t} || \mathcal{R}(x_u, \mathcal{H}_u^{<t}, k) \right].$$

If a pre-existing profile is available in the data, we use it directly. This text is then encoded into a fixed-dimensional user embedding e_u using a frozen sentence embedding model $\text{Enc}(\cdot)$, such that $e_{x_u} = \text{Enc}(p_{x_u}^{<t})$. This embedding serves as a condensed representation of the user's preferences and behavioral patterns.

Position-Aware Input Formulation To enable the hypernetwork to generate distinct parameters for different locations within the LLM, the user embedding e_u is augmented with learnable positional embeddings. For a specific module m at layer l, the input representation $\phi_{x_u}^{m,l}$ is formed by concatenating e_{x_u} with a module embedding $E_{\text{mod}}[m]$ and a depth embedding $E_{\text{dep}}[l]$:

$$\phi_{x_u}^{m,l} = \left[e_{x_u} \,||\, E_{\text{mod}}[m] \,||\, E_{\text{dep}}[l] \right].$$

Parameter Generation The position-aware representation $\phi_u^{m,l}$ is passed through the hypernetwork f_θ, which is implemented as an MLP. The hypernetwork outputs a flattened parameter vector that is then reshaped and split to form the low-rank LoRA matrices, $A_u^{m,l}$ and $B_u^{m,l}$, expressed as

$$(A_{x_u}^{m,l}, B_{x_u}^{m,l}) = \text{Unflatten}(f_\theta(\phi_{x_u}^{m,l})).$$

This process is batched over all target positions $(m, l) \in \mathcal{I}$, where $\mathcal{I}$ is the set of all selected LoRA modules. We denote the complete set of generated parameters for user u as $\Delta W_{x_u} = \text{Gen}_\theta(p_{x_u}^{<t})$.

3.2 Training and Inference

We optimize the hypernetwork parameters θ in an end-to-end fashion across a diverse population of training users. For each user, we use their profile $p_{x_u}^{<t}$ to generate a full set of PEFT parameters. The objective is to minimize the supervised fine-tuning loss on the user's subsequent interactions $\mathcal{H}_u^{\geq t}$ when these parameters are applied to the base model Ψ. The training objective is formulated as

$$\mathcal{L}(\theta) = \mathbb{E}_{u \sim \mathcal{U}} \left[\mathcal{L}_{\text{SFT}} \left(\Psi \oplus \text{Gen}_\theta(p_{x_u}^{<t}), \mathcal{H}_u^{\geq t} \right) \right].$$

By training on a wide variety of users and personalization tasks, the hypernetwork f_θ learns a generalized mapping from natural language user profile to personalized PEFT parameters.

At deployment, the trained hypernetwork enables highly efficient personalization. For any user, including those unseen during training $u \notin \mathcal{U}$, their profile p_u is passed through the generator Gen_θ to produce personalized weights ΔW_u in a single inference pass. This on-demand generation obviates the need for per-user fine-tuning, facilitating scalable and real-time LLM personalization. Furthermore, this framework enhances user privacy; when deployed locally, the hypernetwork can process on-device user data without transmitting sensitive information to external servers. This results in a personalization system that is efficient, scalable, privacy-preserving, and continuously adaptive.

4 Experiment Settings

Baselines We compare P2P against non-personalized base model, retrieval-augmented generation (RAG) (Salemi et al., 2024b), profile-augmented generation (PAG) (Richardson et al., 2023), full user history as context for generation, multi-task LoRA (MT-LoRA), and One PEFT Per User (OPPU) (Tan et al., 2024b).[1] OPPU directly train personal PEFT on the test user history, which can be envisioned as oracle performance. MT-LoRA is trained without user context, and is considered as task-level adaptation without personalization. For all retrieval operations, we use BM25 (Trotman et al., 2014) for efficiency and a fair comparison, and set the number of retrieved items to 2 by default. RAG, PAG, and Full History are prompt-based method fine-tuning, while MT

[1] Please see baseline details in Appendix C.

Table 1: Main experiment results on the LaMP and LongLaMP benchmarks under the *Random split* setting. ↑ indicates that higher values are better, and ↓ implies lower values are preferable. For each task, the best score is in **bold** and the second best is underlined. The final row reports average per-instance inference time (ms).

Task	Metric	Base Model	RAG	PAG	Full History	MT LoRA	OPPU	P2P (Ours)
LaMP-1:	Acc ↑	.519	.504	.563	.562	.511	.531	.583
Citation Id.	F1 ↑	.516	.409	.560	.551	.507	.531	.580
LaMP-2N:	Acc ↑	.653	.666	.761	.750	.670	**.781**	.716
News Cat.	F1 ↑	.679	.679	.773	.764	.683	**.782**	.711
LaMP-2M:	Acc ↑	.345	.351	.372	.413	.386	.391	**.442**
Movie Tagging	F1 ↑	.292	.328	.359	.384	.336	.359	**.408**
LaMP-3:	MAE ↓	.452	.553	.371	.344	.398	**.281**	.383
Product Rating	RMSE ↓	.801	1.00	.699	.724	.731	**.617**	.670
LaMP-4:	R-1 ↑	.121	.130	.128	.144	.150	**.167**	.160
News Headline Gen.	R-L ↑	.108	.118	.116	.131	.135	**.148**	.145
LaMP-5:	R-1 ↑	.465	.474	.436	.472	.473	.468	**.490**
Scholarly Title Gen.	R-L ↑	.404	.414	.382	.412	.409	.422	**.431**
LaMP-7:	R-1 ↑	.376	.378	.378	.407	.382	.353	**.442**
Tweet Paraphrase	R-L ↑	.324	.325	.326	.353	.332	.305	**.437**
LongLaMP-1:	R-1 ↑	.267	.319	.331	.326	.288	**.341**	.314
Abstract Gen.	R-L ↑	.155	.180	.181	.185	.165	**.195**	.177
LongLaMP-2:	R-1 ↑	.283	.267	**.292**	.274	.248	.208	.284
Topic Writing	R-L ↑	.129	.128	.134	.132	.122	.115	**.135**
LongLaMP-3:	R-1 ↑	.212	.240	.308	.235	.231	266	**.247**
Review Writing	R-L ↑	.121	.128	**.145**	.128	.120	.143	.137
Average Performance								
Classification	Acc ↑	.505	.507	.565	.575	.522	.568	**.580**
	F1 ↑	.496	.472	.564	**.566**	.509	.557	**.566**
Generation	R-1 ↑	.287	.301	.312	.310	.295	.301	**.322**
	R-L ↑	.207	.216	.214	.224	.214	.221	**.244**
Infer. Time	ms ↓	31.97	44.58	66.85	461.83	30.51	35.82	39.98

LoRA, OPPU, and P2P request training, we set r to 8 in LoRA for fair comparison. For all methods, we use `Qwen2.5-7B-Instruct` (Yang et al., 2024) as the base model and `Qwen3-Emb-4B` (Zhang et al., 2025b) as the embedding model.

Datasets We employ LaMP (Salemi et al., 2024b), LongLaMP (Kumar et al., 2024), Personal Reddit (PR) (Staab et al., 2023), and Empathetic Conversation (EC) (Omitaomu et al., 2022) datasets in experiments.[2] LaMP consists of three classification tasks (citation identification, movie tagging, news categorization), one rating prediction task, and three text generation tasks (news headline, scholarly title, tweet paraphrasing) and LongLaMP consists of three long-form generation tasks (abstraction generation, topic writing, review writing). All LaMP and LongLaMP tasks contain per-user behavior history, along with query inputs and ground-truth outputs. Empathetic Conversation consists of essay responses based on the article, Personal Reddit consists of Reddit posts. In PR and EC, each user has a textual profile describing user demographic information and personality traits, as well as user inputs and user-written outputs.

Evaluation Settings To assess generalization, we evaluate our model under two data splits: *random* and *out-of-distribution (OOD)*. For the random split, we sample 200 users from each task as the test set; if a task has fewer than 1000 users, we use a standard 80%/20% train-test split. For the OOD split, we construct a challenging test set of users most dissimilar to the training population. To do this, we encode each user's profile into an embedding via `Qwen3-Emb-4B`, perform kmeans clustering, and select smaller and isolated clusters as test set. The OOD test set size mirrors that of the random split for each task, while all remaining

[2]Please see task details in Appendix D.

users are used as training set.[3]

Evaluation Metrics We use task-appropriate metrics to measure performance. For classification, we report Accuracy and F1-score. For text generation, we use ROUGE-1 and ROUGE-L (Lin, 2004), while for rating prediction, we use RMSE and MAE. For the open-ended generation tasks (Empathetic Conversation, Personal Reddit), we employ a LLM judge (`GPT-4o`) to assess personalization quality. We adopt the Prometheus prompt (Kim et al., 2024) with the user profile and rates the output on a 1-5 scale *w.r.t* the user preferences.

5 Results

Tables 1 and 2 present our main results on the LaMP and LongLaMP benchmarks, while Table 3 details the LLM-as-a-Judge evaluations for the PR and EC datasets. The key findings are as follows.

P2P outperforms prompt-based and PEFT-based baselines. Across the majority of tasks in the random split setting (Table 1), P2P demonstrates superior or highly competitive performance. On average, it achieves the highest accuracy in classification tasks (0.577) and the best ROUGE-L scores in generation tasks (0.244). For instance, in the Tweet Paraphrasing task, P2P achieves an ROUGE-1 score of 0.442, significantly outperforming the next best method Full History (0.407). Compared to the PEFT-based OPPU baseline, which requires expensive per-user training, P2P achieves better average performance in both classification and generation without any user-specific fine-tuning at deployment. This highlights its ability to effectively generate high-quality personalized parameters in a single forward pass.

P2P generalizes well to OOD users. The framework demonstrates strong generalization to out-of-distribution users (Table 2). In this challenging split, P2P consistently outperforms other parameter-based baselines like MT-LoRA and OPPU. It also achieves competitive performance against strong prompt-based methods such as PAG and Full History, but with the critical advantage of significantly faster inference and better user privacy preservation. By encoding user history into compact PEFT parameters rather than placing it in the prompt, P2P avoids the substantial computational overhead of processing long contexts with every query and exposing user data to the centralized LLM. The outperformance over OPPU is particularly noteworthy because OPPU is fine-tuned directly on the target user's history, whereas P2P generates parameters without any user-specific training. This suggests that P2P effectively distills collaborative knowledge from the diverse training population, indicating that learning a generalizable mapping from profiles to personalized parameters is a more robust and efficient strategy.

P2P excels at open-ended personalized generation. For tasks requiring nuanced, open-ended generation, LLM-as-a-Judge evaluations (Table 3) confirm the effectiveness of P2P. On both the Personal Reddit and Empathetic Conversation datasets, P2P consistently achieves the highest scores from the `GPT-4o` judge, surpassing both task-level MT-LoRA and prompt-based PAG baselines. Interestingly, combining task-level MT-LoRA adaptation with a RAG approach (appending the user profile to the prompt) does not improve performance under this evaluation paradigm, indicating that these methods struggle to generalize for personalized preference adaptation in open-ended scenarios. Moreover, P2P's robust performance across random and OOD settings demonstrates that its generated parameters effectively capture the stylistic and personal nuances crucial for generating high-quality and personalized responses.

6 Analysis

Training User Quantity *v.s.* Diversity We investigate how P2P's performance varies with training user quantity and diversity. Figure 3 displays performance on *random* and *OOD* splits, varying user diversity by controlling user clusters from 10 to 50 and user count from 20% to 100%. Results indicate that increasing user quantity yields only marginal gains, with top-row performance curves remaining largely flat from 20% to 100% across all tasks. In contrast, increasing diversity positively impacts performance, as bottom-row curves show consistent improvements from 10 to 50 clusters across all task categories for both splits. For example, in classification tasks, OOD F1-score rises from approximately 0.508 to 0.560 with greater diversity, but shows no corresponding gain with increased quantity. These findings suggest that user profile diversity is more critical than sheer training user volume for robust, generalizable performance.

[3]Additional details and statistics are in Appendix H and I.

Table 2: Main experiment results on the LaMP and LongLaMP benchmarks under *OOD split* setting. ↑ indicates that higher values are better, and ↓ implies lower values are preferable. For each task, the best score is in **bold** and the second best is underlined. The final row reports average per-instance inference time (ms).

Task	Metric	Non-Personalized	RAG	PAG	Full History	MT LoRA	OPPU	P2P (Ours)
LaMP-1: Personalized Citation Identification	Acc ↑	.568	.494	**.600**	.592	.561	.556	.576
	F1 ↑	.569	.419	**.600**	.584	.561	.554	.577
LaMP-2N: Personalized News Categorization	Acc ↑	.579	.615	.623	**.655**	.600	.558	.624
	F1 ↑	.601	.639	.630	**.665**	.611	.552	.612
LaMP-2M: Personalized Movie Tagging	Acc ↑	.449	.494	.464	.479	.447	.471	**.543**
	F1 ↑	.406	.483	.461	.454	.410	.416	**.497**
LaMP-3: Personalized Product Rating	MAE ↓	.465	.594	.378	.290	.410	**.198**	.258
	RMSE ↓	.789	1.07	.700	.741	.732	**.540**	.583
LaMP-4: Personalized News Headline Gen.	R-1 ↑	.164	.184	.176	.198	.184	.200	**.210**
	R-L ↑	.140	.164	.154	.174	.165	.180	**.190**
LaMP-5: Personalized Scholarly Title Gen.	R-1 ↑	.455	.473	.448	.495	.470	.468	**.481**
	R-L ↑	.401	.417	.392	**.433**	.426	.422	.431
LaMP-7: Personalized Tweet Paraphrasing	R-1 ↑	.392	.438	.449	**.479**	.393	.379	.473
	R-L ↑	.331	.385	.404	**.424**	.339	.323	.411
LongLaMP-1: Abstract Generation	R-1 ↑	.267	.312	.323	.321	.288	**.326**	.301
	R-L ↑	.153	.177	.175	.179	.164	**.187**	.174
LongLaMP-2: Topic Writing	R-1 ↑	.284	.279	**.287**	.280	.268	.209	.284
	R-L ↑	.129	.135	.136	**.142**	.130	.116	.134
LongLaMP-3: Review Writing	R-1 ↑	.204	.232	**.292**	.234	.170	.247	.220
	R-L ↑	.114	.125	**.140**	.124	.104	.130	.123
Average Performance								
Classification	Acc ↑	.532	.534	.562	.575	.536	.528	**.581**
	F1 ↑	.525	.513	.563	**.567**	.527	.507	.563
Generation	R-1 ↑	.294	.319	.329	**.334**	.295	.305	.326
	R-L ↑	.211	.234	.234	**.246**	.221	.226	.243
Infer. Time	ms ↓	20.52	36.44	61.66	392.97	21.91	26.78	28.64

Table 3: Average LLM-as-a-Judge evaluation scores (on a 1-5 scale) for the Personal Reddit and Empathetic Conversation datasets, comparing performance on *Random* and *OOD* test splits.

Method	Personal Reddit		Empathetic Conv.	
	Random	*OOD*	*Random*	*OOD*
Base Model	1.71	1.58	1.86	1.55
PAG	1.77	1.60	1.80	1.54
MT-LoRA	1.98	1.96	1.62	1.43
P2P (Ours)	**2.21**	**2.15**	**2.03**	**1.65**

Table 4: Performance of P2P with different embedding models on classification, generation, and rating prediction tasks under OOD split.

Embedding Model	Class. ↑		Text Gen. ↑		Rating Pred. ↓	
	Acc	F1	R-1	R-L	MAE	RMSE
`Qwen3-Emb-0.6B`	.562	.544	.316	**.244**	**.258**	**.543**
`Qwen3-Emb-4B`	**.581**	**.562**	**.326**	.243	**.258**	.583
`Qwen3-Emb-8B`	.560	.544	.313	.234	.391	.715
`Qwen2.5-7B-It`	.571	.552	.311	.241	.281	.596
`gte-large-en`	.557	.538	.313	.240	.276	.600
Non-Personalized	.532	.525	.294	.211	.465	.789

Deployment Efficiency We compare the time required to generate personalized PEFT parameters for each user at deployment. On average, OPPU (LoRA) takes 20.44 s per user, OPPU (IA3) takes 22.67 s per user, and OPPU (Prompt Tuning) takes 18.78 s per user. In contrast, our proposed P2P requires only 0.57 s per user, representing a speedup of 33x compared to the fastest OPPU variant. Figure 4 visually confirms this scalability: the cumulative time for OPPU methods increases steeply and linearly with the user count , while the cost for P2P remains near-zero and constant. Although P2P requires a one-time upfront training cost of 27167 seconds, this investment is amortized after 1,450 users, making the framework substantially more efficient for large-scale, real-time applications.

On Embedding Model Choice The quality of user embeddings is critical for the hypernetwork. Our experiments show that all tested embedding backbones yield substantial improvements over the non-personalized baseline across all task categories. Among the dedicated embedding models, our default choice, `Qwen3-Emb-4B`, achieves the

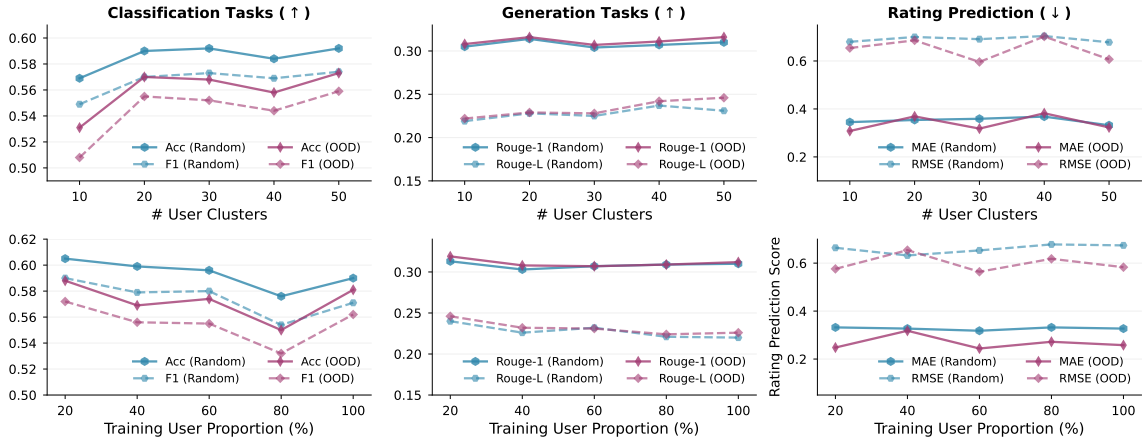

Figure 3: Performance of P2P as a function of training user diversity (top row) and quantity (bottom row). While greater diversity boosts performance, increasing the number of users yields no significant gains.

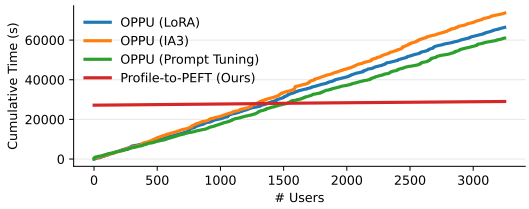

Figure 4: Cumulative time for personalized parameter generation vs. user count. Our Profile-to-PEFT exhibits near-constant, minimal cost, showing superior scalability over OPPU variants' linear growth.

highest classification and text generation scores. Notably, the larger `Qwen3-Emb-8B` model underperforms its smaller counterparts across all metrics, suggesting that simply increasing embedding model size does not guarantee better performance. The framework also proves effective when using the base model's own last-layer activations (`Qwen2.5-7B-It`) as user embeddings. These results demonstrate that while the framework is robust to different embedding backbones, a high-quality, mid-sized model like `Qwen3-Emb-4B` provides the strongest overall performance.

Performance *w.r.t.* User Active Levels User engagement varies significantly. We analyze P2P's robustness across users with varying history lengths in three representative tasks under classification, generation, and rating prediction categories (Shown in Figure 5). The results demonstrate that P2P consistently delivers strong performance regardless of the user's activity level. For all three categories, P2P remains highly competitive with OPPU, which is fine-tuned directly on target user data, and outperforms other baselines across all history length buckets. Even for users with very sparse histories, P2P effectively generates high-quality parameters,

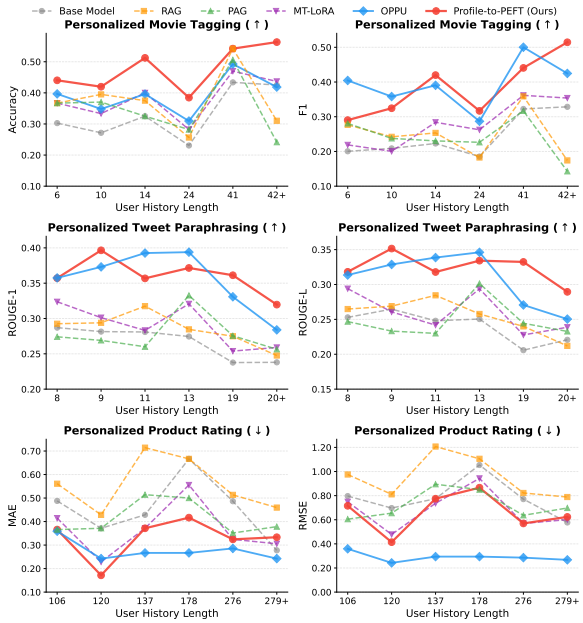

Figure 5: Robustness of P2P to varying user activity levels. The model maintains strong performance across different history lengths, outperforming baselines for both sparse and dense user data.

showcasing its robustness and suitability for real-world scenarios with diverse user engagement.

Ablation Study We conducted systematic ablations to validate key design choices in P2P and show results in Table 5. Disrupting the personalization signal by providing a shuffled user profile causes a significant performance drop across all tasks, with the F1-score for classification falling from 0.562 to 0.521 and the MAE for rating prediction increasing from 0.258 to 0.322. This confirms that the hypernetwork effectively learns from the semantic content of the user profile rather than merely fitting to its structure. Further analysis of the user profile components reveals that the user summary

Table 5: Component ablation study for our method under the *OOD* split, validating their effectiveness.

Embedding Model	Class. ↑		Text Gen. ↑		Rating Pred. ↓	
	Acc	F1	R-1	R-L	MAE	RMSE
P2P (Full)	**.581**	**.562**	**.326**	**.243**	**.258**	**.583**
Personalization Signal Ablations:						
- random user profile	.570	.553	.304	.228	.276	.601
- shuffle user profile	.535	.521	.307	.223	.322	.692
User Profile Contribution:						
- user summary only	.562	.545	.313	.240	.304	.584
- retrieved history only	.538	.521	.298	.216	.405	.712
- full history only	.541	.526	.302	.217	.392	.740

is the most critical input. Using the summary alone achieves performance nearly on par with the full model (*e.g.*, 0.562 *vs.* 0.581 in classification accuracy). In contrast, relying solely on retrieved user history leads to a substantial decline in performance, particularly in rating prediction, where the MAE increases by over 56% (from 0.258 to 0.405). These results underscore the importance of a high-quality user summary as the primary signal for generating effective personalized parameters.

7 Related Work

Personalization of LLMs Existing LLM personalization methods can be broadly categorized into prompt-based and Parameter-Efficient Fine-Tuning (PEFT)-based approaches. Prompt-based methods integrate user data into the model's input context. This includes using raw user history as few-shot examples (Dai et al., 2023; Wang et al., 2024; Kang et al., 2023; Kim and Yang, 2025), retrieving relevant history snippets to overcome context limitations (Salemi et al., 2024b; Mysore et al., 2023; Salemi et al., 2024a), or augmenting queries with summarized user profiles (Richardson et al., 2023; Sun et al., 2025; Dong et al., 2024; Tan et al., 2025b; Zhang, 2024). Further research has explored enhancing these methods with planning (Salemi and Zamani, 2025) and reasoning capabilities (Salemi et al., 2025). In contrast, PEFT-based methods embed user preferences directly into lightweight model parameters. Notable examples include training one PEFT per user (OPPU) (Tan et al., 2024b), enabling collaborative personalization (Tan et al., 2024a), or performing group-level adaptation (Zhang et al., 2025a). User-LLM (Ning et al., 2025) learn user embeddings to contextualize LLMs for personalization. Another line of work focuses on personalized alignment through techniques like parameter merging (Jang et al., 2023), RLHF (Li et al., 2024; Park et al., 2024), custom reward models (Cheng et al., 2023; Bose et al., 2025; Shenfeld et al., 2025), and under black-box model (Zhuang et al., 2024), and conversational settings (Zhao et al., 2025; Wu et al., 2025).

Hypernetwork for PEFT Generation A hypernetwork, a neural network that generates weights for another model (Ha et al., 2016), has become a key strategy for PEFT of LLMs. This approach produces task-specific modules without costly full-model retraining. Pioneering work like HyperFormer used a hypernetwork to generate adapter layers for different NLP tasks (Mahabadi et al., 2021). This concept has since been adapted for various LLM PEFT methods. For instance, some techniques generate soft prompts (He et al., 2022), while others create adapter weights from task embeddings (Phang et al., 2023) or textual descriptions (Ivison et al., 2023). More recent methods focus on generating LoRA parameters directly. While HyperLoRA conditions on few-shot examples (Lv et al., 2024), subsequent approaches like Text-to-LoRA (Charakorn et al., 2025) and DnD (Liang et al., 2025) generate LoRA weights from natural language task descriptions or unlabeled prompts. This evolution enables efficient, on-the-fly adaptation and zero-shot generalization to new tasks without requiring explicit examples or task IDs.

While prior work uses hypernetworks for task-level adaptation, P2P pioneers this approach for user-level personalization. It generates PEFT parameters directly from natural language user profiles or histories, enabling generalization to unseen users and real-time adaptation without per-user fine-tuning. This offers a practical and scalable solution for industrial PEFT-based LLM personalization.

8 Conclusion

We introduce P2P, a hypernetwork-based framework that generates personalized PEFT parameters directly from user profiles. This approach produces customized LoRA adapters in a single inference pass at deployment, eliminating the costly per-user fine-tuning of traditional methods and enabling real-time updates. Experiments demonstrate that P2P matches the performance of computationally intensive baselines and generalizes effectively and robustly to unseen users during deployment. By decoupling parameter generation from per-user training, P2P offers a practical path toward deploying dynamic, privacy-preserving, and truly individualized LLMs at scale, paving the way for instantly adaptive personalized AI systems.

Limitations

We identify two key limitations in P2P. First, constrained by the dataset, our focus is primarily on one specific task per user rather than examining user behaviors across multiple tasks and domains. For instance, in the movie tagging task, users are solely engaged in that specific activity, without the inclusion of behaviors from other domains or platforms. Despite this, the P2P framework is inherently adaptable to any text sequence generation task and is compatible with diverse user instructions across various tasks and domains. Personalizing LLM across a broader range of tasks and domains is left as future work. Second, despite our proposed P2P is compatible with all PEFT methods that introduce trainable modules throughout the model, such as Adapter (Houlsby et al., 2019), $(\mathrm{IA})^3$ (Liu et al., 2022), and prefix tuning (Li and Liang, 2021), we primarily focus on LoRA in this work. This is due to LoRA's popularity, widespread use, and superior performance demonstrated by OPPU (Tan et al., 2024b), while we expect to expand our experiment and analysis to more PEFT methods in future work.

Ethical Considerations

Privacy The Profile-to-PEFT framework is designed to enhance user privacy by enabling local deployment, where personalized parameters are generated without transmitting raw user history to a central server. This significantly mitigates the risk of data leakage during transmission. However, the generated PEFT parameters themselves are a compressed representation of a user's profile and preferences. This raises a potential concern that the parameters could be reverse-engineered to infer sensitive information about the user. Therefore, ensuring the security and privacy of these generated adapter weights is crucial, especially if they are stored or managed by a service provider.

Data Bias and Manipulation The personalization process inherently relies on a user's historical data. If this data contains existing biases, stereotypes, or prejudices, the hypernetwork will learn to encode these undesirable patterns into the personalized PEFT parameters. This could lead to the LLM generating outputs that reinforce or amplify a user's biases, creating a harmful echo chamber. Furthermore, the ability to directly map a profile to model behavior could be exploited for malicious manipulation, where a crafted profile could generate parameters that cause the LLM to subtly persuade or mislead a user. It is essential to develop methods for auditing and mitigating bias in both the input user data and the resulting personalized models.

Accessibility and Responsibility While Profile-to-PEFT significantly lowers the computational barrier for deploying personalized models compared to the "One-PEFT-Per-User" approach, the initial training of the hypernetwork on a diverse user population remains a resource-intensive task. This could still present an accessibility challenge for smaller organizations or researchers, potentially concentrating the power to create such systems in the hands of a few large entities. Developers of this technology have a responsibility to consider these downstream effects and to implement safeguards that prevent the system from being used for harmful purposes, such as generating discriminatory content or facilitating large-scale manipulation.

References

Avinandan Bose, Zhihan Xiong, Yuejie Chi, Simon Shaolei Du, Lin Xiao, and Maryam Fazel. 2025. Lore: Personalizing llms via low-rank reward modeling. *arXiv preprint arXiv:2504.14439*.

Dorian Brown. 2020. Rank-BM25: A Collection of BM25 Algorithms in Python.

Rujikorn Charakorn, Edoardo Cetin, Yujin Tang, and Robert Tjarko Lange. 2025. Text-to-lora: Instant transformer adaption. *CoRR*, abs/2506.06105.

Jin Chen, Zheng Liu, Xu Huang, Chenwang Wu, Qi Liu, Gangwei Jiang, Yuanhao Pu, Yuxuan Lei, Xiaolong Chen, Xingmei Wang, Kai Zheng, Defu Lian, and Enhong Chen. 2024. When large language models meet personalization: perspectives of challenges and opportunities. *World Wide Web (WWW)*, 27(4):42.

Pengyu Cheng, Jiawen Xie, Ke Bai, Yong Dai, and Nan Du. 2023. Everyone deserves a reward: Learning customized human preferences. *Preprint*, arXiv:2309.03126.

Sunhao Dai, Ninglu Shao, Haiyuan Zhao, Weijie Yu, Zihua Si, Chen Xu, Zhongxiang Sun, Xiao Zhang, and Jun Xu. 2023. Uncovering chatgpt's capabilities in recommender systems. In *Proceedings of the 17th ACM Conference on Recommender Systems, RecSys 2023, Singapore, Singapore, September 18-22, 2023*, pages 1126–1132. ACM.

Yijiang River Dong, Tiancheng Hu, and Nigel Collier. 2024. Can LLM be a personalized judge? In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 10126–10141, Miami, Florida, USA. Association for Computational Linguistics.

Jian Guan, Junfei Wu, Jia-Nan Li, Chuanqi Cheng, and Wei Wu. 2025. A survey on personalized Alignment—The missing piece for large language models in real-world applications. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 5313–5333, Vienna, Austria. Association for Computational Linguistics.

David Ha, Andrew M. Dai, and Quoc V. Le. 2016. Hypernetworks. *ArXiv*, abs/1609.09106.

Charles R Harris, K Jarrod Millman, Stéfan J Van Der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J Smith, and 1 others. 2020. Array programming with numpy. *Nature*, 585(7825):357–362.

Yun He, Steven Zheng, Yi Tay, Jai Gupta, Yu Du, Vamsi Aribandi, Zhe Zhao, YaGuang Li, Zhao Chen, Donald Metzler, and 1 others. 2022. Hyperprompt: Prompt-based task-conditioning of transformers. In *International conference on machine learning*, pages 8678–8690. PMLR.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799. PMLR.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2021. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Hamish Ivison, Akshita Bhagia, Yizhong Wang, Hannaneh Hajishirzi, and Matthew Peters. 2023. HINT: Hypernetwork instruction tuning for efficient zero- and few-shot generalisation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11272–11288, Toronto, Canada. Association for Computational Linguistics.

Joel Jang, Seungone Kim, Bill Yuchen Lin, Yizhong Wang, Jack Hessel, Luke Zettlemoyer, Hannaneh Hajishirzi, Yejin Choi, and Prithviraj Ammanabrolu. 2023. Personalized soups: Personalized large language model alignment via post-hoc parameter merging. *arXiv preprint arXiv:2310.11564*.

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

Wang-Cheng Kang, Jianmo Ni, Nikhil Mehta, Maheswaran Sathiamoorthy, Lichan Hong, Ed Chi, and Derek Zhiyuan Cheng. 2023. Do llms understand user preferences? evaluating llms on user rating prediction. *Preprint*, arXiv:2305.06474.

Jaehyung Kim and Yiming Yang. 2025. Few-shot personalization of LLMs with mis-aligned responses. In *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)*, pages 11943–11974, Albuquerque, New Mexico. Association for Computational Linguistics.

Seungone Kim, Jamin Shin, Yejin Choi, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoo Yun, Seongjin Shin, Sungdong Kim, James Thorne, and Minjoon Seo. 2024. Prometheus: Inducing fine-grained evaluation capability in language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Hannah Rose Kirk, Bertie Vidgen, Paul Röttger, and Scott A Hale. 2024. The benefits, risks and bounds of personalizing the alignment of large language models to individuals. *Nature Machine Intelligence*, pages 1–10.

Ishita Kumar, Snigdha Viswanathan, Sushrita Yerra, Alireza Salemi, Ryan A Rossi, Franck Dernoncourt,

Hanieh Deilamsalehy, Xiang Chen, Ruiyi Zhang, Shubham Agarwal, and 1 others. 2024. Longlamp: A benchmark for personalized long-form text generation. *arXiv preprint arXiv:2407.11016*.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Cheng Li, Mingyang Zhang, Qiaozhu Mei, Yaqing Wang, Spurthi Amba Hombaiah, Yi Liang, and Michael Bendersky. 2023. Teach llms to personalize–an approach inspired by writing education. *arXiv preprint arXiv:2308.07968*.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597, Online. Association for Computational Linguistics.

Xinyu Li, Zachary C Lipton, and Liu Leqi. 2024. Personalized language modeling from personalized human feedback. *arXiv preprint arXiv:2402.05133*.

Zhiyuan Liang, Dongwen Tang, Yuhao Zhou, Xuanlei Zhao, Mingjia Shi, Wangbo Zhao, Zekai Li, Peihao Wang, Konstantin Schürholt, Damian Borth, and 1 others. 2025. Drag-and-drop llms: Zero-shot prompt-to-weights. *arXiv preprint arXiv:2506.16406*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin A Raffel. 2022. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. *Advances in Neural Information Processing Systems*, 35:1950–1965.

Jiahong Liu, Zexuan Qiu, Zhongyang Li, Quanyu Dai, Jieming Zhu, Minda Hu, Menglin Yang, and Irwin King. 2025. A survey of personalized large language models: Progress and future directions. *arXiv preprint arXiv:2502.11528*.

Chuancheng Lv, Lei Li, Shitou Zhang, Gang Chen, Fanchao Qi, Ningyu Zhang, and Hai-Tao Zheng. 2024. Hyperlora: Efficient cross-task generalization via constrained low-rank adapters generation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 16376–16393.

Rabeeh Karimi Mahabadi, Sebastian Ruder, Mostafa Dehghani, and James Henderson. 2021. Parameter-efficient multi-task fine-tuning for transformers via shared hypernetworks. *arXiv preprint arXiv:2106.04489*.

Sheshera Mysore, Zhuoran Lu, Mengting Wan, Longqi Yang, Steve Menezes, Tina Baghaee, Emmanuel Barajas Gonzalez, Jennifer Neville, and Tara Safavi. 2023. Pearl: Personalizing large language model writing assistants with generation-calibrated retrievers. *arXiv preprint arXiv:2311.09180*.

Lin Ning, Luyang Liu, Jiaxing Wu, Neo Wu, Devora Berlowitz, Sushant Prakash, Bradley Green, Shawn O'Banion, and Jun Xie. 2025. User-llm: Efficient llm contextualization with user embeddings. In *Companion Proceedings of the ACM on Web Conference 2025*, pages 1219–1223.

Damilola Omitaomu, Shabnam Tafreshi, Tingting Liu, Sven Buechel, Chris Callison-Burch, Johannes Eichstaedt, Lyle Ungar, and João Sedoc. 2022. Empathic conversations: A multi-level dataset of contextualized conversations. *arXiv preprint arXiv:2205.12698*.

Chanwoo Park, Mingyang Liu, Kaiqing Zhang, and Asuman Ozdaglar. 2024. Principled rlhf from heterogeneous feedback via personalization and preference aggregation. *arXiv preprint arXiv:2405.00254*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, and 1 others. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Jason Phang, Yi Mao, Pengcheng He, and Weizhu Chen. 2023. Hypertuning: Toward adapting large language models without back-propagation. In *International Conference on Machine Learning*, pages 27854–27875. pmlr.

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

Alireza Salemi, Surya Kallumadi, and Hamed Zamani. 2024a. Optimization methods for personalizing large language models through retrieval augmentation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2024, Washington DC, USA, July 14-18, 2024*, pages 752–762. ACM.

Alireza Salemi, Cheng Li, Mingyang Zhang, Qiaozhu Mei, Weize Kong, Tao Chen, Zhuowan Li, Michael Bendersky, and Hamed Zamani. 2025. Reasoning-enhanced self-training for long-form personalized text generation. *arXiv preprint arXiv:2501.04167*.

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

Alireza Salemi and Hamed Zamani. 2025. Lamp-qa: A benchmark for personalized long-form question answering. *arXiv preprint arXiv:2506.00137*.

Idan Shenfeld, Felix Faltings, Pulkit Agrawal, and Aldo Pacchiano. 2025. Language model personalization via reward factorization. *arXiv preprint arXiv:2503.06358*.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H Chi, Nathanael Schärli, and Denny Zhou. 2023. Large language models can be easily distracted by irrelevant context. In *International Conference on Machine Learning*, pages 31210–31227. PMLR.

Robin Staab, Mark Vero, Mislav Balunovic, and Martin Vechev. 2023. Beyond memorization: Violating privacy via inference with large language models. In *The Twelfth International Conference on Learning Representations*.

Chenkai Sun, Ke Yang, Revanth Gangi Reddy, Yi Ren Fung, Hou Pong Chan, Kevin Small, ChengXiang Zhai, and Heng Ji. 2025. Persona-db: Efficient large language model personalization for response prediction with collaborative data refinement. In *Proceedings of the 31st International Conference on Computational Linguistics, COLING 2025, Abu Dhabi, UAE, January 19-24, 2025*, pages 281–296. Association for Computational Linguistics.

Zhaoxuan Tan and Meng Jiang. 2023. User modeling in the era of large language models: Current research and future directions. *IEEE Data Eng. Bull.*, 47(4):57–96.

Zhaoxuan Tan, Zheng Li, Tianyi Liu, Haodong Wang, Hyokun Yun, Ming Zeng, Pei Chen, Zhihan Zhang, Yifan Gao, Ruijie Wang, Priyanka Nigam, Bing Yin, and Meng Jiang. 2025a. Aligning large language models with implicit preferences from user-generated content. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 7792–7820. Association for Computational Linguistics.

Zhaoxuan Tan, Zheyuan Liu, and Meng Jiang. 2024a. Personalized pieces: Efficient personalized large language models through collaborative efforts. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 6459–6475. Association for Computational Linguistics.

Zhaoxuan Tan, Qingkai Zeng, Yijun Tian, Zheyuan Liu, Bing Yin, and Meng Jiang. 2024b. Democratizing large language models via personalized parameter-efficient fine-tuning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 6476–6491. Association for Computational Linguistics.

Zhaoxuan Tan, Zinan Zeng, Qingkai Zeng, Zhenyu Wu, Zheyuan Liu, Fengran Mo, and Meng Jiang. 2025b. Can large language models understand preferences in personalized recommendation? *CoRR*, abs/2501.13391.

Andrew Trotman, Antti Puurula, and Blake Burgess. 2014. Improvements to bm25 and language models examined. In *Proceedings of the 19th Australasian Document Computing Symposium*, pages 58–65.

Danqing Wang, Kevin Yang, Hanlin Zhu, Xiaomeng Yang, Andrew Cohen, Lei Li, and Yuandong Tian. 2024. Learning personalized alignment for evaluating open-ended text generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 13274–13292. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and 1 others. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45.

Shujin Wu, Yi R. Fung, Cheng Qian, Jeonghwan Kim, Dilek Hakkani-Tur, and Heng Ji. 2025. Aligning LLMs with individual preferences via interaction. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 7648–7662, Abu Dhabi, UAE. Association for Computational Linguistics.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, and 22 others. 2024. Qwen2.5 technical report. *CoRR*, abs/2412.15115.

Jiarui Zhang. 2024. Guided profile generation improves personalization with large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 4005–4016, Miami, Florida, USA. Association for Computational Linguistics.

Linhai Zhang, Jialong Wu, Deyu Zhou, and Yulan He. 2025a. PROPER: A progressive learning framework for personalized large language models with group-level adaptation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16399–16411, Vienna, Austria. Association for Computational Linguistics.

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. 2025b. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *CoRR*, abs/2506.05176.

Zhehao Zhang, Ryan A. Rossi, Branislav Kveton, Yijia Shao, Diyi Yang, Hamed Zamani, Franck Dernoncourt, Joe Barrow, Tong Yu, Sungchul Kim, Ruiyi Zhang, Jiuxiang Gu, Tyler Derr, Hongjie Chen, Ju-Ying Wu, Xiang Chen, Zichao Wang, Subrata Mitra, Nedim Lipka, and 2 others. 2024. Personalization of large language models: A survey. *ArXiv*, abs/2411.00027.

Zheng Zhao, Clara Vania, Subhradeep Kayal, Naila Khan, Shay B. Cohen, and Emine Yilmaz. 2025. Personalens: A benchmark for personalization evaluation in conversational AI assistants. In *Findings of the Association for Computational Linguistics, ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 18023–18055. Association for Computational Linguistics.

Aakas Zhiyuli, Yanfang Chen, Xuan Zhang, and Xun Liang. 2023. Bookgpt: A general framework for book recommendation empowered by large language model. *arXiv preprint arXiv:2305.15673*.

Yuchen Zhuang, Haotian Sun, Yue Yu, Rushi Qiang, Qifan Wang, Chao Zhang, and Bo Dai. 2024. HYDRA: model factorization framework for black-box LLM personalization. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

A Performance *w.r.t.* Retrieval Top k

We analyze the impact of the number of retrieved historical items k on the performance of both P2P and the RAG baseline across classification, generation, and rating prediction tasks. As shown in the Figure 6, the performance of P2P remains remarkably stable and consistently high across all values of k, from 0 to 32. This holds true for both the random and OOD splits. The flat performance curves indicate that our method is not sensitive to the number of retrieved items, suggesting that the user summary provides a strong, condensed personalization signal that makes the model robust and less reliant on a dynamic retrieval step for each input. In contrast, the performance of the RAG baseline is highly dependent on k. For classification and generation tasks, RAG's performance generally improves as more items are retrieved, though it often plateaus or slightly degrades with a very high number of items. This highlights RAG's sensitivity to the quality and quantity of the retrieved context. Across all tasks and splits, P2P consistently outperforms the RAG baseline, demonstrating the superiority of encoding user preferences into the model's parameters over simply augmenting the input prompt.

B Performance with Additional Base Model

To evaluate the robustness of our framework, we replicated our experiments using a smaller base model, `Qwen2.5-3B-it`. The results, presented in Table 6 and 7, demonstrate that the core advantages of P2P are consistent across different model sizes.

Even with the smaller 3B model, P2P consistently outperforms the computationally expensive OPPU baseline and remains highly competitive with strong prompt-based methods in both the *random* and *OOD* splits. In the random split, our method achieves the highest average accuracy (0.565) and F1-score (0.557) in classification tasks. Similarly, in the more challenging OOD split, it maintains a significant lead over other parameter-based methods like OPPU and MT-LoRA, showcasing its strong generalization capabilities.

While the absolute performance scores are naturally slightly lower than those achieved with the larger 7B model, the relative performance gains and overall trends remain the same. This confirms that the effectiveness of the P2P framework is not contingent on a large-scale base model and that our approach provides a robust and scalable solution for LLM personalization.

C Baseline Details

We present the details of baseline methods to help facilitate reproducibility.

- **Retrieval-Augmented Generation (RAG)** (Salemi et al., 2024b): The LLM input is appended with the top k relevant user history items *w.r.t.* the user input x_u, where the input sequence x' can be defined as

$$x'_u = [\mathcal{R}(x_u, \mathcal{H}_u, k) \mid\mid x_u],$$

  where $\mathcal{R}$ is the retriever, default to BM25. $[\cdot||\cdot]$ denotes the concatenation operation.

- **Profile-Augmented Generation (PAG)** (Richardson et al., 2023): The LLM input is appended with user summary s_u and top k retrieved user history items that most relevant to user input x_u. The user summary s_u is generated by the base model from the sampled user history entries. The input of LLM x' is represented as

$$x'_u = [s_u \mid\mid \mathcal{R}(x_u, \mathcal{H}_u, k) \mid\mid x_u].$$

- **Full History**: in this method, the user context is the entire user history corpus. Given the long context window of 32k tokens, the model can consume almost all of the historical data. If the flattened user history overflow the context length, we keep the most recent history items that fit in the context length. The input of LLM x' is represented as

$$x'_u = [\text{Flatten}(\mathcal{H}_u) \mid\mid x_u].$$

- **Multi-task LoRA (MT-LoRA)**: The multi-task LoRA method serves as a task-level adaptation without user context for personalization. It trains a LoRA using the input and output pairs from the training set without including user profile or user history for personalization signals. The training objective can be represented as:

$$\Delta W^* = \arg\min_{\Delta W} \mathcal{L}_{\text{SFT}}(\Psi \oplus \Delta W, \{(x_u, y_u)\}),$$

  where the Ψ is the base model parameters, $\{(x_u, y_u)\}$ denotes the user input and output pair in training set for SFT training.

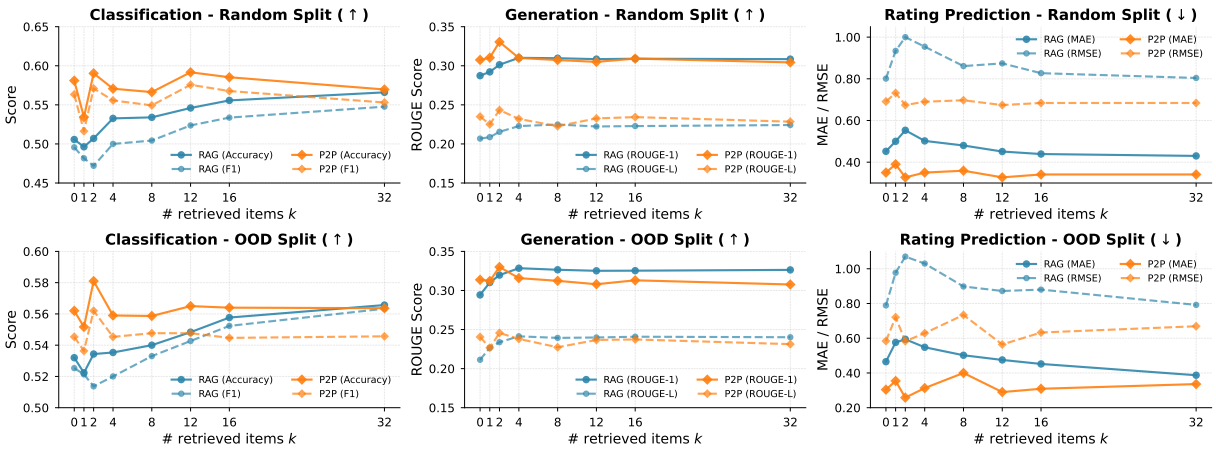

Figure 6: Performance of P2P and RAG baseline method with different retrieval item k under both *Random* and *OOD* split.

- **One PEFT Per User (OPPU)** (Tan et al., 2024b): OPPU trains the per-user PEFT parameters on the target user history from scratch. The personalized PEFT parameters for user u can be represented as:

$$\Delta W_u^* = \arg\min_{\Delta W} \mathcal{L}_{\text{SFT}}(\Psi \oplus \Delta W, \mathcal{H}_u^{<t}),$$

  The ΔW_u requires per-user training using the target user history data from scratch and needs iteratively went through the entire user history corpus using backpropagation to optimize the PEFT parameters, making it request extensive computation as the number of users scale up.

D Task Details

We present the task details as follows to help readers gain a better understanding of the task format.

- **LaMP-1: Personalized Citation Identification** is a binary text classification task. Specifically, given user u writes a paper x, the task aims to make the model determine which of the two candidate papers u will cite in paper x based on the user's history data, which contains the publications of user u.

- **LaMP-2N: Personalized News Categorization** is a 15-way text classification task to classify news articles written by a user u. Formally, given a news article x written by user u, the language model is required to predict its category from the set of categories based on the user's history data, which contains the user's past article and corresponding category.

- **LaMP-2M: Personalized Movie Tagging** is a 15-way text classification task to make tag assignments aligned with the user's history tagging preference. Specifically, given a movie description x, the model needs to predict one of the tags for the movie x based on the user's historical movie-tag pairs.

- **LaMP-3: Personalized Product Rating** is a 5-way text classification task and can also be understood as a regression task. Given the user u's historical review and rating pairs and the input review x, the model needs to predict the rating corresponding to x selected from 1 to 5 in integer.

- **LaMP-4: Personalized News Headline Generation** is a text generation task to test the model's ability to capture the stylistic patterns in personal data. Given a query x that requests to generate a news headline for an article, as well as the user profile that contains the author's historical article-title pairs, the model is required to generate a news headline specifically for the given user.

- **LaMP-5: Personalized Scholarly Title Generation** is a text generation task to test personalized text generation tasks in different domains. In this task, we require language models to generate titles for an input article x, given a user profile of historical article-title pairs for an author.

- **LaMP-7: Personalized Tweet Paraphrasing** is also a text generation task that tests the model's capabilities in capturing the stylistic patterns of authors. Given a user input text x and the user profile of historical tweets, the model is required

Table 6: Main experiment results on the LaMP and LongLaMP benchmarks under the *random split* setting using `Qwen2.5-3B-it` as base model. ↑ indicates that higher values are better, and ↓ implies lower values are preferable. For each task, the best score is in **bold** and the second best is underlined. The final row reports average per-instance inference time (ms).

Task	Metric	Base Model	RAG	PAG	Full History	MT LoRA	OPPU	P2P (Ours)
LaMP-1: Citation Id.	Acc ↑	.456	.543	.574	.582	.511	.480	**.591**
	F1 ↑	.421	.530	.574	.577	.490	.475	**.585**
LaMP-2N: News Cat.	Acc ↑	.544	.630	**.724**	.701	.655	.605	.69
	F1 ↑	.575	.652	**.741**	.719	.644	.635	.683
LaMP-2M: Movie Tagging	Acc ↑	.247	.326	.332	.384	.307	.226	**.413**
	F1 ↑	.262	.349	.344	.397	.298	.244	**.402**
LaMP-3: Product Rating	MAE ↓	.941	.859	.665	.475	.495	.656	**.405**
	RMSE ↓	1.31	1.27	1.08	.889	.882	1.04	**.794**
LaMP-4: News Headline Gen.	R-1 ↑	.119	.137	.129	.147	.141	.132	**.155**
	R-L ↑	.109	.125	.117	.132	.128	.118	**.141**
LaMP-5: Scholarly Title Gen.	R-1 ↑	.443	.474	.423	.468	.454	.449	**.484**
	R-L ↑	.379	.412	.362	.407	.412	.384	**.438**
LaMP-7: Tweet Paraphrase	R-1 ↑	.370	.377	.364	.383	**.466**	.321	.462
	R-L ↑	.316	.322	.308	.323	**.403**	.270	.397
LongLaMP-1: Abstract Gen.	R-1 ↑	.312	**.348**	.337	.347	.343	.304	.346
	R-L ↑	.170	.189	.187	.191	.190	.161	**.200**
LongLaMP-2: Topic Writing	R-1 ↑	.226	.249	**.271**	.246	.251	.237	.204
	R-L ↑	.115	.127	**.134**	.129	.128	.114	.117
LongLaMP-3: Review Writing	R-1 ↑	.191	.233	**.271**	.223	.229	.202	.222
	R-L ↑	.112	.126	**.137**	.124	.129	.113	.131
Average Performance								
Classification	Acc ↑	.415	.499	.543	.556	.491	.437	**.565**
	F1 ↑	.419	.510	.553	**.564**	.477	.451	.557
Generation	R-1 ↑	.277	.303	.299	.302	**.314**	.274	.312
	R-L ↑	.200	.217	.208	.218	.232	.193	**.237**
Infer. Time	ms ↓	21.94	31.58	43.25	258.66	20.86	24.13	27.51

to paraphrase x into y that follows the given user's tweet pattern.

- **LongLaMP-1: Abstract Generation** is a long-form text generation task designed to test personalized summarization. Given a user's history of writing academic papers and the body of a new, unseen paper, the model is required to generate an abstract for the new paper that aligns with the user's personal writing style.
- **LongLaMP-2: Topic Writing** is a long-form text generation task that assesses the model's ability to adopt a user's unique writing voice. Given a specific topic and a user profile containing their past writings, the model must generate a new piece of text on the given topic that emulates the user's personal style.
- **LongLaMP-3: Review Writing** is a long-form generation task focused on capturing a user's personal voice and opinion patterns. Given a product or business and a user's history of past reviews, the model is tasked with generating a new review that reflects the user's characteristic style and rating tendencies.
- **Empathetic Conversation** (Omitaomu et al., 2022): consists of 1000 essay responses (both empathy score and textual response) to a news article with their demographics and self-reported personality traits. It further includes dialog interactions between paired participants, enriched with various dialog annotations, such as other-reported empathy levels and turn-by-turn emotion ratings. This dataset can be used as both multiple-choice question answering and text generation dataset.
- **Personal Reddit** (Staab et al., 2023): consists of 500 samples of Reddit posts with their

Table 7: Main experiment results on the LaMP and LongLaMP benchmarks under the *OOD split* setting using `Qwen2.5-3B-it` as base model. ↑ indicates that higher values are better, and ↓ implies lower values are preferable. For each task, the best score is in **bold** and the second best is underlined. The final row reports average per-instance inference time (ms).

Task	Metric	Non-Personalized	RAG	PAG	Full History	MT LoRA	OPPU	P2P (Ours)
LaMP-1: Personalized Citation Identification	Acc ↑	.491	.560	.580	**.588**	.505	.502	.573
	F1 ↑	.455	.555	**.580**	.586	.491	.502	.570
LaMP-2N: Personalized News Categorization	Acc ↑	.437	.550	.544	.583	.519	.495	**.596**
	F1 ↑	.471	.582	.567	**.591**	.505	.522	.582
LaMP-2M: Personalized Movie Tagging	Acc ↑	.271	.376	.374	**.476**	.327	.203	.430
	F1 ↑	.308	.419	.419	**.507**	.338	.239	.432
LaMP-3: Personalized Product Rating	MAE ↓	1.00	.802	.571	.327	.387	.635	**.239**
	RMSE ↓	1.40	.126	.979	.715	.803	.107	**.599**
LaMP-4: Personalized News Headline Gen.	R-1 ↑	.151	.184	.172	.188	.179	.151	**.193**
	R-L ↑	.133	.164	.153	.168	.158	.134	**.175**
LaMP-5: Personalized Scholarly Title Gen.	R-1 ↑	.450	.467	.434	**.477**	.461	.454	.476
	R-L ↑	.395	.407	.383	.421	.420	.397	**.432**
LaMP-7: Personalized Tweet Paraphrasing	R-1 ↑	.384	.441	.447	.449	**.483**	.358	.453
	R-L ↑	.320	.384	.389	.395	**.406**	.299	.382
LongLaMP-1: Abstract Generation	R-1 ↑	.306	.340	.336	**.342**	.331	.295	.330
	R-L ↑	.164	.188	.185	.184	.183	.156	**.189**
LongLaMP-2: Topic Writing	R-1 ↑	.228	.251	**.274**	.240	.225	.232	.206
	R-L ↑	.115	.135	**.140**	.132	.120	.111	.119
LongLaMP-3: Review Writing	R-1 ↑	.175	.223	**.261**	.213	.209	.195	.204
	R-L ↑	.102	.121	**.133**	.119	.118	.107	.120
Average Performance								
Classification	Acc ↑	.400	.495	.499	**.549**	.450	.400	.533
	F1 ↑	.411	.518	.522	.451	.445	.421	**.528**
Generation	R-1 ↑	.282	.318	**.321**	.318	.315	.280	.310
	R-L ↑	.205	.233	.231	**.236**	.234	.201	**.236**
Infer. Time	ms ↓	22.17	50.59	33.76	223.24	22.43	24.91	26.83

(anonymized) personal attributes, such as location, income, and sex. It contains user profile, question, and a ground-truth response given by the user, which can be used in our SFT training.

E Experimental Details

To promote task diversity during P2P training, each batch contains four different personalization tasks, with sampling weights proportional to the square root of each task's dataset size to limit oversampling of small datasets. We train P2P with a learning rate of 2×10^{-5} for 20,000 steps and a batch size of 32 by default. For the generated LoRA adapters, we set the rank r=8 and insert trainable parameters into `q_proj` and `v_proj`. For inference, we use greedy decoding with temperature $\tau = 0$ to reduce randomness and improve reproducibility.

F Computational Resources

All the training experiments in this paper were conducted on a single node with 8 × NVIDIA A100-SXM4-80GB GPUs and Intel(R) Xeon(R) Platinum 8275CL CPU @ 3.00GHz.

G Scientific Artifacts

P2P is built with the help of many existing scientific artifacts, including PyTorch (Paszke et al., 2019), Numpy (Harris et al., 2020), rank-bm25 (Brown, 2020), and huggingface transformers (Wolf et al., 2020). We use vllm (Kwon et al., 2023) as the inference framework. We will make the P2P implementation publicly available to facilitate further research.

H Dataset Splits Details

To ensure a representative random test split, we adopt a clustering-based diverse user selection strategy that leverages user embeddings to capture underlying population structures. Specifically, K-means clustering is applied to the normalized embeddings, with the number of clusters adaptively set between a minimum and maximum based on

Table 8: Detailed statistics for all dataset splits. For each file, we report the number of users, number of queries, and the average length of input contexts and output generations (in character).

Dataset	Subtask	Split	Users	Queries	Avg. Input Len	Avg. Output Len
LaMP	Citation Id.	train	5947	7289	355.25	3.00
		ood_test	200	255	377.62	3.00
		random_test	200	254	358.93	3.00
	Movie Tagging	train	733	5511	602.91	9.76
		ood_test	93	409	602.64	7.62
		random_test	92	518	602.92	9.36
	News Cat.	train	253	7721	457.15	9.47
		ood_test	32	1106	445.29	8.90
		random_test	32	823	449.61	9.53
	News Headline Gen.	train	1396	12147	178.02	62.50
		ood_test	100	487	219.65	49.77
		random_test	100	1125	174.10	63.62
	Product Rating	train	19244	21658	694.97	1.00
		ood_test	200	217	493.73	1.00
		random_test	200	221	739.47	1.00
	Scholarly Title Gen.	train	14274	15738	1094.14	76.60
		ood_test	200	218	1156.29	74.27
		random_test	200	217	1065.66	75.35
	Tweet Paraphrase	train	12912	14346	180.44	93.16
		ood_test	200	225	184.37	93.75
		random_test	200	223	178.26	92.32
LongLaMP	Abstract Generation	train	22103	30950	261.17	1116.49
		ood_test	200	283	256.51	1098.07
		random_test	200	277	259.04	1106.19
	Product Review	train	15490	18928	725.36	1652.50
		ood_test	200	256	745.18	1813.55
		random_test	200	251	740.06	1412.50
	Topic Writing	train	15330	19897	196.90	1438.58
		ood_test	200	279	184.61	1410.62
		random_test	200	273	208.48	1381.02
Personal Reddit	-	train	409	409	394.66	656.11
		ood_test	53	53	385.45	602.70
		random_test	52	52	376.73	590.27
Empathetic Conv.	-	train	53	712	4555.78	402.21
		ood_test	10	153	4307.86	487.34
		random_test	10	109	4438.62	385.30

dataset size to achieve balanced group sizes. Cluster proportions are computed, and users are sampled proportionally from each cluster using random selection within clusters, ensuring the selected subset mirrors the overall distribution while incorporating randomness for variability; adjustments are made for rounding errors and small clusters to meet the exact target user count. In contrast, for the OOD test split, an extreme clustering approach is employed to maximize dissimilarity from the training population. The optimal number of clusters is determined via silhouette score maximization over a searched range, after which clusters are analyzed for size, intra-cluster compactness (average distance to centroid), and inter-cluster isolation (average distance to other centroids). A scoring metric prioritizing small, isolated clusters (defined as the inter-cluster distance divided by (1 + cluster size) guides the selection: entire high-scoring (outlier) clusters are allocated to the OOD test if they fit within the target size, or the farthest users from the centroid are chosen otherwise, enforcing strict cluster separation such that no training users originate from OOD-assigned clusters, thereby enhancing the OOD set's distinctiveness.

I Dataset Statics

The dataset statistics are presented in Table 8.

J Prompt Details

We present the prompt template for user profile generation and LLM-as-a-Judge evaluation.

Reward Scoring Prompt Template

###Task Description:
An instruction (might include an Input inside it), a response to evaluate, a reference answer that can get a score of 5, a user profile containing user preferences and information, and a score rubric representing a evaluation criteria are given.
1. Write a detailed feedback that assess the quality of the response strictly based on the given score rubric, not evaluating in general.
2. Consider how well the response aligns with the user's preferences, interests, and background information provided in the user profile when evaluating personalization quality.
3. After writing a feedback, write a score that is an integer between 1 and 5. You should refer to the score rubric.
4. The output format should look as follows: "(write a feedback for criteria) [RESULT] (an integer number between 1 and 5)"
5. Please do not generate any other opening, closing, and explanations.

###The instruction to evaluate:
{{ instruction }}

###Response to evaluate:
{{ response }}

###Reference Answer (Score 5):
{{ reference_answer }}

###User Profile:
{{ user_profile }}

###Score Rubrics:
{{ rubric }}

###Feedback:

Score Rubric

`criteria`:"Evaluate how well the response to the instruction is personalized to the specific user."
`score1_description`: "Generic or impersonal. Ignores the provided profile/personality. Style does not match the user; may feel robotic or off-topic. Makes incorrect assumptions or contradicts stated preferences. No meaningful use of user details; largely boilerplate."
`score2_description`: "Minimal personalization. Mentions a profile detail superficially but remains mostly generic. Weak style match; limited relevance to the user's interests or situation. Includes filler or distracting disclaimers. Significant deviation from the reference's intent or emphasis."
`score3_description`: "Basic personalization. References a few relevant details and partially adapts tone. Generally on topic but misses important user nuances (interests, constraints, or personality cues). Moderate similarity to the reference; may be verbose or somewhat generic."
`score4_description`: "Good personalization. Integrates multiple user details accurately; content is relevant and helpful. Tone largely matches the user's personality and preferred style. Clear, concise, and engaging with only minor misses versus the user's preferences or the reference's intent."
`score5_description`: "Excellent personalization. Seamlessly weaves in pertinent profile details; highly relevant and tailored guidance or conversation. Tone precisely matches the user's personality—empathetic, engaging, and concise. Avoids boilerplate and unnecessary disclaimers. Closely aligned with the user's likely preference as indicated by the reference."

Rubric Template

[`{criteria}`]
Score 1: `{score1_description}`
Score 2: `{score2_description}`
Score 3: `{score3_description}`
Score 4: `{score4_description}`
Score 5: `{score5_description}`

User Profile Generation Prompt

Instruction
Generate a targeted user profile for {{ task_description }} based on the provided user history data. This profile will be used to understand user behavior patterns specific to this task.
IMPORTANT: You must analyze HOW this user behaves and makes decisions relevant to this task, NOT list WHAT specific content they interact with. Do not output lists of topics, keywords, or content examples. Focus only on behavioral tendencies, preferences, and decision-making patterns relevant to the target task.
Focus on understanding patterns that inform task-specific user behavior:
1. Task-Relevant User Preferences:
- Decision-making patterns and criteria relevant to the target task
- Quality and content preferences that influence choices
- Style and approach preferences in task-related activities
- Consistency patterns in task-related decision making
2. Behavioral Patterns Related to Task Performance:
- Interaction patterns and engagement styles relevant to the task
- Response patterns to different types of content or options
- Timing and frequency patterns in task-related activities
- Adaptation patterns when encountering new or different scenarios
3. Personal Style and Voice Indicators:
- Communication style patterns relevant to the task
- Personal expression tendencies and voice characteristics
- Authenticity markers and personal touch preferences
- Consistency in personal style across different contexts
4. Context Awareness and Adaptation Patterns:
- Awareness of audience, context, or requirements in task-related activities
- Adaptation strategies for different scenarios within the task domain
- Personalization approaches and individual preference integration
- Balance between task requirements and personal style
User History Data: {{ user_history }}
Output Format
Output the user profile strictly in plain text describing the user's behavioral patterns, preferences, and decision-making tendencies specifically relevant to {{ task_description }}. Focus on patterns that predict how this user would approach and perform the target task.
Do NOT output:
- Lists of topics, keywords, or content they engage with
- Specific examples of content, products, or interactions
- Names of people, places, brands, or entities
- Content examples or subject matter details
DO output:
- Behavioral patterns and preferences relevant to the task
- Decision-making tendencies and criteria
- Personal style and approach patterns
- Task-specific interaction patterns
Derive insights strictly from the provided historical data. Do not include explanations, introductions, headings, bullet points, or any formatting structure.

Task Description Prompt

`LaMP-1 Citation Identification`: 'Academic citation recommendation: Identify relevant reference papers for researchers based on their publication titles and research focus areas.'

`LaMP-2N News Categorization`: 'News article categorization: Classify news articles into topical categories based on content, subject matter, and thematic focus.'

`LaMP-2M Movie Tagging`: 'Movie genre tagging: Analyze movie descriptions and assign appropriate genre tags based on content themes, narrative elements, and stylistic features.'

`LaMP-3 Product Rating`: 'Product review rating prediction: Analyze product reviews and predict the rating score based on sentiment, content quality, and expressed satisfaction levels.'

`LaMP-4 News Headline Generation`: 'News article headline generation: Create a concise and engaging headline for news articles based on their content and key themes.'

`LaMP-5 Scholarly Title Generation`: 'Academic title generation: Generate concise and descriptive titles for research papers based on abstracts, capturing the main research contribution and scope.'

`LaMP-7 Tweet Paraphrasing`: 'Tweet paraphrasing: Rewrite tweets in a personal style while maintaining the original meaning and adapting the tone and language to individual communication patterns.'

`LongLaMP-1 Abstract Generation`: 'Academic abstract generation: Create comprehensive abstracts for research papers based on titles and key research items, incorporating domain-specific knowledge and writing style.'

`LongLaMP-2 Topic Writing`: 'Topic-based content generation: Create personalized Reddit posts on given topics that reflect individual writing style, interests, and communication preferences.'

`LongLaMP-3 Review Writing`: 'Product review generation: Write detailed product reviews that reflect personal experiences, preferences, and writing style based on ratings and product features.'

`Empathetic Conversation`: 'Empathetic news commentary: Provide personal commentary and emotional reactions to news articles that reflect individual values, perspectives, and empathetic responses.'

`Personal Reddit`: 'Personal conversation response: Generate authentic and personalized responses to conversations that reflect individual personality, background, and communication style.'

User Profile Embedding Prompt

Instruction:
Extract and represent key personalization features that reflect the user's unique characteristics, preferences, and behavioral patterns from the provided user information. Focus on demographics, stated and implicit preferences, interaction history, behavioral trends, and other traits that can inform task-specific personalization. Pay special attention to features relevant for the following task: {{ task_description }}

User Information:
{{ user_profile }}

LaMP-1 Citation Identification Prompt

Instruction:
Identify the most relevant reference for the listed publication by the researcher. Select the reference paper that is most closely related to the researcher's work.

Paper Title:
{paper_title}

Options:
{options_str}

Output Format:
Just answer with [1] or [2] without explanation.

Answer:

LaMP-2N News Categorization Prompt

Instruction:
Which category does this article relate to among the following categories? Just answer with the category name without further explanation.

Categories:
travel, education, parents, style & beauty, entertainment, food & drink, science & technology, business, sports, healthy living, women, politics, crime, culture & arts, religion
Article:
{article}

Output Format:
Just answer the category without further explanation.

Answer:

LaMP-2M Movie Tagging Prompt

Instruction:
Which tag does this movie relate to among the following tags?

Tags:
sci-fi, based on a book, comedy, action, twist ending, dystopia, dark comedy, classic, psychology, fantasy, romance, thought-provoking, social commentary, violence, true story

Description:
{description}

Output Format:
Just answer with the tag name without further explanation.

Answer:

LaMP-3 Product Rating Prediction Prompt

Instruction:
What is the score of the following review on a scale of 1 to 5?

Review:
{review_text}

Output Format:
Just answer with 1, 2, 3, 4, or 5 without further explanation.

Answer:

LaMP-4 News Headline Generation Prompt

Instruction:
Generate a headline for the following article.

Article:
{article}

Output Format:
Just answer the headline without further explanation.

Answer:

LaMP-5 Scholarly Title Generation Prompt

Instruction:
Generate a title for the following abstract of a paper.

Abstract:
{abstract}

Output Format:
Just answer the title without further explanation.

Answer:

LaMP-7 Tweet Paraphrasing Prompt

Instruction:
Paraphrase the following text into tweet without any explanation before or after it.

Original Tweet:
{original_tweet}

Output Format:
Just answer the paraphrased tweet without further explanation.

Answer:

LongLaMP-1 Abstract Generation Prompt

Instruction:
Given the paper title and key items, generate an abstract for the paper.

Paper Title:
{paper_title}

Output Format:
Only output the abstract. Do not include any explanation or formatting.

Answer:

LongLaMP-2 Topic Writing Prompt

Instruction:
Generate the content for a Reddit post based on the provided topic.

Topic:
{topic_prompt}

Output Format:
Only output the reddit post. Do not include any explanation or formatting.

Answer:

LongLaMP-3 Review Writing Prompt

Instruction:
Given the overall rating, product description, and review summary, generate the review text written by a reviewer.

Overall Rating:
{rating}

Product Description:
{product_description}

Review Summary:
{review_summary}

Output Format:
Only output the review text. Do not include any explanation or formatting.

Answer:

Emphathetic Conversation Prompt

Article:
{article}

Instruction:
After reading the news article, write an essay (300-800 characters) summarizing your thoughts and feelings about the article.

Answer:

K Qualitative Examples

We present qualitative examples in Table 9 and 10.

Table 9: User Profile and Generation Example from LaMP-2: Movie Tagging task.

User Profile	This user tends to categorize and tag movies with a strong emphasis on plot elements rather than stylistic elements. They show a consistent tendency to label movies with tags such as "dark comedy," "comedy," and "dystopia," indicating a preference for narratives that blend humor with darker themes or explore societal issues. The user also demonstrates a clear sensitivity to specific movie genres like "sci-fi," "action," and "violence," often, indicating they enjoys them the user association and thematic elements and focusing user system tends to categorize movies movies movies based on these genres. They tend to recognize and tag hybrid genres, such as "dark comedy," showing an ability to identify films that blend multiple thematic elements. In terms of genre recognition, the user shows a preference for mainstream tags over niche or artistic labels. They frequently use tags like "classic," "based on a book," and "fantasy," suggesting an appreciation for films that have cultural significance or are derived from literary works. The user exhibits a critical analysis depth in tag assignment, often choosing precise tags that reflect the complexity of the film's narrative or thematic content. They demonstrate a higher tagging confidence in genres they are knowledgeable about, such as "comedy" and "action." Demographic influences play a role in the user's tag perception and assignment. They seem to prefer tags that align with their personal tastes, such as "comedy" and "dark comedy," while also recognizing the broader appeal of mainstream genres like "action" and "sci-fi." Overall, this user's tagging behavior indicates a focus on narrative structure and thematic complexity, with a strong preference for mainstream genres and a tendency to use precise, descriptive tags that reflect the film's content and style.
User Input	Peter Parker is an outcast high schooler abandoned by his parents as a boy, leaving him to be raised by his Uncle Ben and Aunt May. Like most teenagers, Peter is trying to figure out who he is and how he got to be the person he is today. As Peter discovers a mysterious briefcase that belonged to his father, he begins a quest to understand his parents' disappearance 2013 leading him directly to Oscorp and the lab of Dr. Curt Connors, his father's former partner. As Spider-Man is set on a collision course with Connors' alter ego, The Lizard, Peter will make life-altering choices to use his powers and shape his destiny to become a hero.
Non-Personalized Output	based on a book (✗)
RAG Output	based on a book (✗)
PAG Output	based on a book (✗)
OPPU	comedy (✗)
P2P Output	action (✓)

Table 10: User Profile and Generation Example from LaMP-7: Tweet Paraphrasing task.

User Profile	This user tends to write in a casual and emotive style, often using informal language and expressing strong feelings. They frequently use contractions and colloquialisms, such as "totly" and "yup". Their tone can shift between affectionate ("awwwwww", "lol") and self-deprecating humor ("bitch", "like an abandoned child"). The user also shows a preference for sharing personal experiences and emotions, indicating a tendency to personalize content on personal on their on social media ands user. The user frequently uses abbreviations and emojis for emphasis, suggesting an expressive and engaging communication style. They tend to share personal details and express vulnerability, which could indicate a preference for authenticity and openness in their interactions. However, they also seem to value privacy, as seen in their reluctance to engage in conversations when feeling disturbed.In terms of paraphrasing, this user is likely to retain the core emotion and context of a tweet while personalizing the language. They may use more casual and emotionally charged words, incorporate personal anecdotes, and maintain a conversational tone. Their approach to paraphrasing suggests a focus on maintaining the original meaning through relatable and authentic expressions.
Original Tweet	@msnye, I hope you pray for me because there's a chance I might end up at the Emergency Room tonight.
Reference Answer	@msnye u better.... Man o man pray for me cause I might end up @ the ER before the night is over.
Non-Personalized Output	@msnye Need your prayers, might head to ER tonight
RAG Output	@msnye I hope you're praying for me as I might head to the ER tonight.
PAG Output	@msnye, I hope you're prayin' for me 'cuz I might hit the ER tonight
OPPU Output	@msnye Need your prayers, thinking of heading to ER tonight. Fingers crossed!
P2P Output	@msnye please pray for me tonight, I might be in the ER