

Preference Descriptions for Dynamic Personalization of Large Language Models

Naofumi Osawa

DENTSU SOKEN INC., X(Cross) Innovation Division, AI Transformation Center
2-17-1, Konan, Minato-ku, Tokyo 108-0075, Japan
osawa.naofumi@dentsusoken.com

Abstract

Personalization of large language models (LLMs) has recently gained attention as a key approach to enhancing their usability and adaptability to individual users. However, when personalization relies on interaction histories, inconsistencies often arise between past and current user preferences, causing the model to fail to properly adapt to the user’s most recent intent.

We propose a simple preference representation that encodes each preference as an **intensity** and **polarity** vector and **aggregates** past and recent preferences arithmetically to form a inconsistency-free, up-to-date personalized prompt.

This allows the LLM to dynamically update user preferences at test time, linearly combining past and current preferences to produce a consistent, inconsistency-free personalized prompt. The proposed method does not require explicit ground truth data and can be applied to various downstream applications. Preliminary experiments using datasets demonstrate that our approach improves preference alignment compared to a non personalized model and achieves dynamic optimization over time.

Introduction

Large Language Model (LLM) based agents have evolved beyond simple conversational systems into autonomous problem solvers capable of integrating external tools and performing long-term, goal directed reasoning. These agents have shown high utility in a wide range of domains, including dialog systems, recommendation engines, and personal assistants. However, most existing LLM agents still rely on a uniform “one-size-fits-all” paradigm, lacking the flexibility to adapt to individual user preferences and contexts.

Personalization is essential for providing reliable and engaging user experiences. Previous studies have explored several personalization strategies, such as fine tuning, retrieval augmented methods that utilize external information, and persona based prompting. While these approaches have achieved certain improvements, they still face the following limitations:

1. **Coarse preference representation:** User preferences are often treated as static or categorical, failing to capture the degree or variation of polarity and intensity.

2. **Lack of temporal adaptivity:** Personalization depends heavily on the initial setup and cannot dynamically respond to changes in user preferences during interactions.
3. **Practical inefficiency:** Frequent parameter updates or large-scale retrieval processes make these methods unsuitable for scalable, real world applications.

Recent research (Zhang et al. 2025a) has proposed frameworks that combine episodic and semantic memory with dynamically optimized system prompts. While these approaches represent important progress, they still lack explicit modeling of preference polarity (positive/negative) and intensity, and they offer limited mechanisms for handling temporally continuous preference fluctuations.

To address these challenges, we propose a **continuous personalization framework** that leverages both long-term and short-term memory. Specifically, our framework introduces:

- **Preference Description** that encodes polarity and intensity, enabling nuanced modeling of user inclinations and their strength;
- **Prompt Initialization**, which extracts such preferences from past interactions to initialize the agent;
- **Continuous Alignment**, which continuously updates preference representations during interactions to reflect dynamic user changes.

Contribution

We applied our approach to multiple datasets and observed consistent improvements in preference alignment and adaptability compared to non personalized baselines. The key contributions of this work are summarized as follows:

1. **Preference Vector Representation:** We introduce a novel numerical representation of user preferences that explicitly models both *polarity* (positive/negative inclination) and *intensity* (preference strength). This formulation enables continuous and fine-grained personalization, capturing subtle variations in user tendencies over time.
2. **Dynamic Personalization Framework:** We propose an integrated framework combining **Prompt Initialization** and **Continuous Alignment**, allowing large language models to dynamically update user preference representations at test time. This mechanism maintains temporal

consistency and prevents contradictions between long-term and recent user behaviors.

3. **Empirical Validation and Scalability:** Through comprehensive experiments on benchmark datasets such as *ImplicitPersona* and *PersonaMem*, we demonstrate that our method significantly improves preference alignment and response adaptability compared to non personalized and prompt engineering baselines. These results highlight the scalability and general applicability of the proposed framework across diverse personalization tasks.

Related Work

Personalization Method

There are multiple patterns for personalizing LLMs.(Zhang et al. 2025b; Liu et al. 2025) For personalization, various types of information can be considered, such as the user’s own profile, documents created by the user, and logs of interactions with the system.(Samarati and Sweeney 1998; Ni, Li, and McAuley 2019; Dinan et al. 2019; Jang et al. 2023; Kirk et al. 2024) In this paper, we propose a personalization approach that utilizes interaction logs from already in use LLM applications.

Preference Polarity & Intensity Modeling

Beyond treating preferences as static categories, we model each preference by *polarity* (dislike/neutral/like) and *intensity* (strength). This design is inspired by established practices in computational social science: sentiment frameworks that score *valence with intensity* (e.g., VADER) (Hutto and Gilbert 2014) and the use of Best–Worst Scaling to obtain reliable, fine-grained strength judgments. We adapt these ideas from sentiment to preference representation by defining $p_j \in \{-1, 0, 1\}$ (polarity) and $i_j \in \{0, \dots, 5\}$ (intensity), and aggregating them arithmetically to maintain up-to-date, inconsistency-free preference states during interaction.

Evaluation of Personalization

Evaluation of personalization includes text generation approaches that assess similarity between LLM generated and human generated content, recommendation and classification systems that evaluate the ability to suggest user preferences(Salemi et al. 2024; Lin 2004; Banerjee and Lavie 2005) and recently approaches using LLM as a Judge.(Gu et al. 2025; Zheng et al. 2023) Text generation and recommendation systems require acquiring human preferences as a gold standard beforehand, which poses challenges in real world operations.

Personalization of LLMs for Application

Recently, personalization of large language models (LLMs) has attracted significant attention. For example, ChatGPT and Claude have introduced memory features (OpenAI 2025b; Anthropic 2024), enabling persistent adaptation across sessions. In the research community, personalization has been studied from multiple angles: (i) incorporating user descriptions and histories for personalized recommendations, (ii) augmenting LLMs with external long-term

memory to carry context across interactions, and (iii) establishing benchmarks for systematic evaluation.

In education, context personalization has been shown to improve learner motivation by generating vocabulary learning examples tailored to individual interests (Leong et al. 2024). In recommender systems, LLMs can act as interactive preference interfaces, showing competitive performance near cold start settings using only natural language preference statements (Sanner et al. 2023). From the model architecture perspective, LongMem extends LLMs with long-term memory by caching past key-value states in a non differentiable memory and retrieving them through a side network, yielding improvements in long context modeling and many shot in-context learning (Wang et al. 2023).

Proposed Method

Episodic Memory

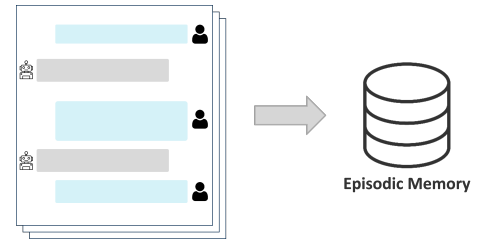


Figure 2: Construction of Episodic Memory

Drawing inspiration from *PersonaAgent* (Zhang et al. 2025a), we conceptualize the interaction log with the system as an instance of Episodic Memory (Tulving 1972). Accordingly, we define \mathcal{D} as follows. The Episodic Memory is designed to record when the user engaged in a conversation, the topic under discussion, and the corresponding system response.

$$\mathcal{D}^u = \{(q_i, r_i, Aux_i)\}_{i=1}^{N^u}, \quad (1)$$

In this formulation, q_i denotes a user query, r_i denotes the response generated by the LLM Application, Aux_i encapsulates auxiliary meta information such as the execution timestamp or a unique identifier, and N^u represents the total number of interaction records. Episodic memory is employed both for the construction of the Preference Description (see the following subsection) and as a retrieval augmented generation (RAG) mechanism that enables the LLM, via function calling, to search and retrieve past user interactions.

Preference Description

Preference Description is a memory module inspired by Semantic Memory (Tulving 1972), designed to manage a user’s preferences along with their polarity and intensity. It consists of descriptions that represent the user’s consistent traits and preferences, together with the polarity and strength of those preferences. When discrepancies arise between past Preference Descriptions and the user’s most recent ones, arithmetic addition is performed to resolve the inconsistency. The generation of the Preference Description

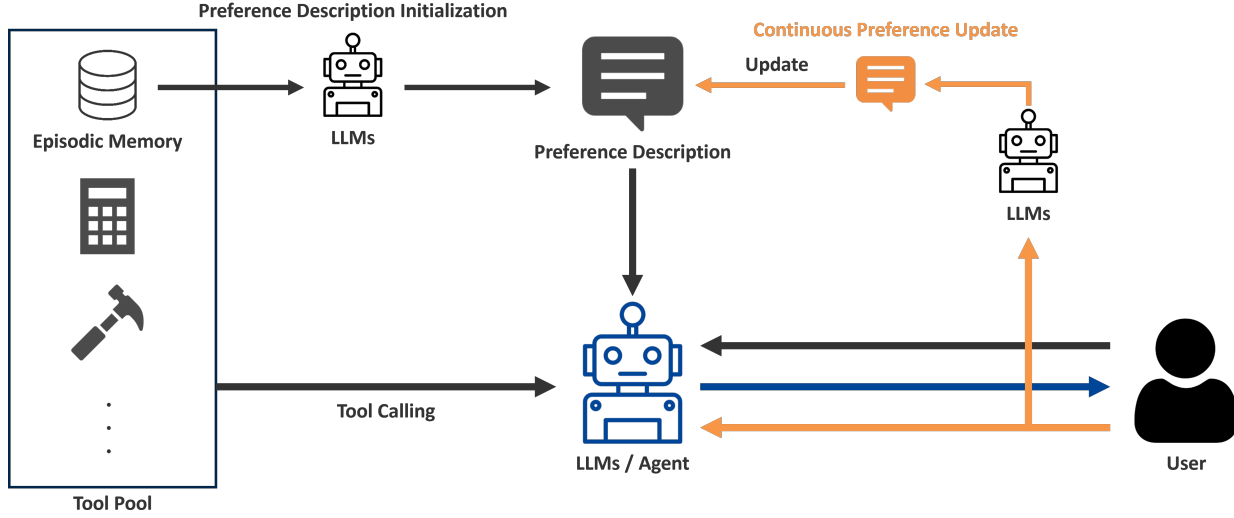


Figure 1: Proposed Method Architecture

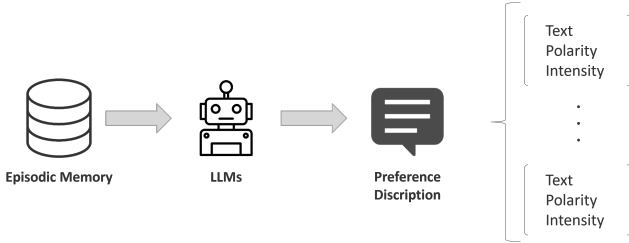


Figure 3: Construction of Preference Description

utilizes a summarization function f_s based on a Large Language Model (LLM), which is defined as follows:

$$\mathcal{L}^u = \{(t_j, i_j, p_j)\}_{j=1}^J = f_s(S_t, \mathcal{D}^u) \quad (2)$$

where

- t_j : preference text,
- $i_j \in \{0, 1, 2, 3, 4, 5\}$: intensity level,
- $p_j \in \{-1, 0, 1\}$: polarity.

And J is a variable depending on the description generation process. S_t is summarization prompt to extract user preferences (Appendix A). Generated Preference Description is injected into the system prompt of the LLM or agent to optimize the LLM’s behavior for the user.

Continuous Preference Update

In the proposed method, a long-term memory component called the *Interaction Log* is used to generate a *Preference Description*, which serves to construct a *Personalized LLM*. In real-world scenarios, during testing, it is necessary to continuously optimize the LLM for individual users through ongoing interactions. To achieve this, we propose an alignment approach that utilizes the short-term memory—that is, the most recent interactions with the LLM—to reflect the

user’s preferences at test time. The key feature of the proposed method is the use of *intensity* and *polarity* to combine the Preference Description, which represents past user preferences, with recent user tendencies without contradiction, thereby constructing the most optimal and up-to-date user preference representation. By continuously updating the preference representation, our approach achieves a balance between short-term adaptability and long-term stability, enabling the system to effectively follow the temporal changes in user preferences (*Concept Drift*). Moreover, since the preference vector is dynamically refined through interactions, the dependence on the initial Preference Description is reduced, allowing the personalized model to gradually converge toward the user’s true and evolving preferences. For example, if a user previously disliked spicy food but has recently become able to enjoy it, our approach enables the system to recommend appropriately spiced foods that align with the user’s current preference level, rather than excessively spicy foods. The algorithm for updating the preferences is presented below, and the concatenated prompt S_c is described in Appendix B.

Algorithm 1: Continuous Preference Update

Input: Test-time user data $\mathcal{D}_{\text{test}}$, initial Preference Description $\mathcal{L}_{\text{init}}$

Output: Updated Preference Description \mathcal{L}^*

- 1: Partition $\mathcal{D}_{\text{test}}$ into batches $\{\mathcal{D}_{\text{batch}}\}$
 - 2: $\mathcal{L}^* \leftarrow \mathcal{L}_{\text{init}}$
 - 3: **for** each $\mathcal{D}_{\text{batch}} \subseteq \mathcal{D}_{\text{test}}$ **do**
 - 4: $\Delta_{\text{batch}} \leftarrow f_s(S_t, \mathcal{D}_{\text{batch}})$
 - 5: $\mathcal{L}^* \leftarrow \text{LLM}(S_c, \mathcal{L}^*, \Delta_{\text{batch}})$
 - 6: **end for**
 - 7: **return** \mathcal{L}^*
-

Experiments

We conducted preliminary experiments to verify the effectiveness of the proposed method. This section describes the datasets, evaluation methods, and experimental results. The experiments aimed to confirm the following two aspects:

1. Personalization through Preference Description.
2. Personalization in continuous dialog via Continuous Preference Update.

Experiment 1: Personalization through Preference Description

In this experiment, we used the evaluation metric called PHS, as defined below, to assess whether the LLM using the Preference Description generated by our proposed method can appropriately reflect user preferences in its responses. We employed multiple OpenAI models for evaluation (OpenAI 2025a, 2024), and fixed the reasoning effort of the reasoning model to *medium* during all experiments.

Dataset For this experiment, we used the bowen-upenn/ImplicitPersona dataset (Jiang et al. 2025). This dataset includes dialog histories (`chat_history`) and user persona information, and has been applied to question-answering and text generation tasks. In our experiment, we generated Preference Description from the `chat_history` field, produced responses to the `user_query` using the proposed method, and evaluated how well the responses were optimized for each user using the PHS score described below.

Baselines We compared our proposed method with two baselines: (1) the system prompt construction method of PersonaAgent (Zhang et al. 2025a), and (2) a non personalized approach.

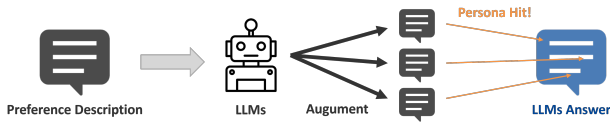


Figure 4: Persona Hit Score

Evaluation Metrics : PHS In evaluating text generation by large language models (LLMs), various metrics such as BLEU, ROUGE, METEOR, and BERTScore have been widely used to measure the semantic similarity between generated texts and reference texts (Chang et al. 2024). While these metrics are effective for quantifying the naturalness and semantic coherence of generated outputs, they cannot directly assess the extent to which a user’s persona information (e.g., preferences or attributes) is reflected in the responses. Moreover, in contexts requiring personalized or user adaptive generation, open ended generation without reference texts is common, making conventional reference based metrics less suitable for evaluation.

To address this issue, we propose the **Persona Hit Score (PHS)**, a novel evaluation metric based on the **LLM-as-a-Judge** framework. PHS utilizes an LLM to assess how well

the user’s preferences are reflected in the response across the following four dimensions:

- **Relevance**
- **Coverage**
- **Specificity**
- **Consistency**

These four dimensions were selected to comprehensively capture the alignment between the generated response and the user’s persona information. *Relevance* measures whether the response addresses the user’s preferences or interests; *Coverage* assesses how broadly the response incorporates various aspects of the user’s described preferences; *Specificity* evaluates whether the reflection of preferences is expressed in a concrete and personalized manner; and *Consistency* examines whether the expressed persona remains coherent across the entire response. Together, these criteria enable a balanced and interpretable evaluation of persona alignment in generated texts.

By leveraging the LLM-as-a-Judge approach, the proposed method flexibly interprets user preferences described in the Preference Description and evaluates whether these preferences are appropriately incorporated into the generated responses.

The detailed evaluation prompts are provided in Appendix C.

Results The results of Experiment 1 are shown in Table 1. We evaluated the proposed method by changing several models for both the LLM and the LLM as a Judge. As a result, the proposed method achieved the highest PHS score compared to the other methods in all LLM models. In addition, the PHS score tended to increase proportionally with the performance of the LLM model. From these results, it can be concluded that the proposed method effectively retains user preferences and reflects them in response generation, outperforming the other approaches. The BERTScore-F1 was calculated by comparing each generated response with the `correct_answer` provided in the dataset. However, there was no significant difference observed across methods or LLM models.

Experiment 2: Personalization in continuous dialog via Continuous Preference Update

In the second experiment, we conducted continuous dialog experiments using the bowen-upenn/PersonaMem dataset (Jiang et al. 2025) to evaluate whether the proposed method can generate consistent and contextually appropriate responses in multi turn conversations. We used a dataset with a 128k-token context and performed updates with larger batch sizes, setting $D_{\text{batch}} \in \{50, 100, 150, 200\}$ in Algorithm 1. By comparing the generated responses with the ground truth, we confirmed that the proposed method successfully updates the Preference Description continuously, thereby reflecting the user’s most recent preferences. In this experiment, we used a Retrieval Augmented Generation (RAG) approach to obtain the Episodic Memory of the proposed method, using the `large` text embedding model 3.

Dataset	Metrics	Model	Ours	w/o intensity,polarity	Non-Personalization
bowen-upenn/ImplicitPersona	PHS(GPT-4.1)	GPT-5	85.1	82.9	38.8
	PHS(GPT-5)	GPT-5	85.4	83.4	50.9
	PHS(GPT-4.1)	GPT-4.1	84.6	84.3	26.0
	PHS(GPT-5)	GPT-4.1	76.5	76.3	36.4
	PHS(GPT-4.1)	GPT-4o	60.6	52.6	21.7
	PHS(GPT-5)	GPT-4o	55.3	50.2	30.5
	BERTScore-F1	GPT-5	0.80	0.80	0.80
	BERTScore-F1	GPT-4.1	0.83	0.82	0.82
	BERTScore-F1	GPT-4o	0.83	0.83	0.83

Table 1: Comparison of the average PHS scores of the proposed method and baselines, including preference text only and non personalized models, on general QA tasks. The PHS scores were obtained by running the LLM five times and averaging the results.

The top 5 results retrieved from the RAG process were provided as reference inputs to the LLM. We also employed multiple LLM(OpenAI, Anthropic, and Google) models for evaluation, and fixed the reasoning effort of the reasoning model to *medium* during all experiments.

Dataset We used the bowen-upenn/PersonaMem dataset (Jiang et al. 2025) to conduct continuous multi turn dialogs. We verified that the proposed method could generate consistent and appropriate responses without contradictions during multi turn conversations. Specifically, we used the Acknowledge latest user preferences subset of the dataset and evaluated the correctness of the model responses by measuring how accurately the answers generated by our method matched the ground truth.

Baselines We compared our method with two baselines: the system prompt construction approach of PersonaAgent (Zhang et al. 2025a), and a non personalized approach without user adaptation.

Evaluation Metrics Based on the accuracy defined below, we evaluated the responses generated by the LLM with our proposed method by comparing its selected answers with the ground truth.

$$\text{Accuracy} = \frac{\sum_{i \in \mathcal{D}_{\text{dataset}}} \mathbf{1}(S_{\text{llm}}^{(i)} = S_{\text{ground truth}}^{(i)})}{|\mathcal{D}_{\text{dataset}}|}, \quad (3)$$

where $\mathcal{D}_{\text{dataset}}$ denotes the evaluation dataset.

Results The experimental results are shown in Table 2. Our proposed method, which continuously updates the Preference Description, achieved higher accuracy in selecting

user preferred responses compared to the baselines. Interestingly, the proposed method performed better with non Reasoning models than with the Reasoning model. In addition, the comparative method that excluded *intensity* and *polarity* from the Preference Description also achieved relatively high accuracy, suggesting that fine-grained expressions based on intensity and polarity were not necessarily effective for generating the user’s desired responses.

Discussion

In Experiment 1, we proposed and evaluated the LLM-as-a-Judge approach to assess how effectively user attributes and preferences are reflected in the responses generated by LLMs. Compared with PersonaAgent and the non personalized approach, our proposed method most accurately captured user preferences. These results reaffirm the effectiveness of prompt optimization that explicitly incorporates user preferences into the system prompt, and further demonstrate that leveraging *intensity* and *polarity* enables more fine-grained representations of user preferences. However, since the PHS score is a newly defined metric introduced in this study, it is necessary to further validate its reliability by comparing it with human evaluations and by testing it in real world scenarios.

Although our proposed method achieved higher persona alignment in terms of the Persona Hit Score (PHS), the BERTScore-F1 did not show a notable difference across models (Table 1). This behavior can be attributed to the intrinsic characteristics of BERTScore, which evaluates token level semantic similarity and is less sensitive to structural or stylistic divergence.

In our experiments, the `correct_answer` in datasets typically consisted of concise, single sentence suggestions

Dataset	Approach	Model	$D_{batch}=50$	100	150	200
bowen-upenn/PersonaMem	Ours	GPT-5	0.71	0.75	0.74	0.74
		GPT-4.1	0.72	0.77	0.76	0.79
		Claude Sonnet 4.5	0.84	0.80	0.78	0.83
		Claude Opus 4.5	0.79	0.79	0.79	0.78
		Claude Haiku 4.5	0.58	0.56	0.57	0.64
		Gemini 2.5 Pro	0.78	0.78	0.75	0.80
	w/o RAG	GPT-5	0.49	0.57	0.53	0.57
		GPT-4.1	0.65	0.61	0.64	0.68
		Claude Sonnet 4.5	0.66	0.68	0.67	0.75
		Claude Opus 4.5	0.63	0.69	0.64	0.72
		Claude Haiku 4.5	0.45	0.43	0.48	0.48
		Gemini 2.5 Pro	0.67	0.69	0.68	0.74
	w/o RAG, intensity, polarity	GPT-5	0.52	0.57	0.55	0.60
		GPT-4.1	0.65	0.65	0.68	0.67
		Claude Sonnet 4.5	0.66	0.68	0.69	0.73
		Claude Opus 4.5	0.62	0.68	0.65	0.71
		Claude Haiku 4.5	0.43	0.43	0.49	0.48
		Gemini 2.5 Pro	0.63	0.72	0.70	0.74
	Non-Personalization	GPT-5	0.09			
		GPT-4.1	0.21			
		Claude Sonnet 4.5	0.32			
		Claude Opus 4.5	0.19			
		Claude Haiku 4.5	0.13			
		Gemini 2.5 Pro	0.39			
Random Guess		0.25				

Table 2: Averaged results across multiple runs on varying \mathcal{D}_{batch} sizes using different personalization approaches.

(e.g., “you might enjoy starting your morning with gentle yoga and gratitude journaling”), whereas our model generated responses provided richer and more diverse expressions in enumerated or reflective formats (e.g., a list of mindful routines, personalized examples, and contextually adaptive advice). Although these outputs semantically covered the same core ideas, the surface level structure diverged significantly. Because BERTScore aggregates contextual embeddings over all tokens, such semantic overlap maintains a high similarity score even when the generated text differs in form, verbosity, or organization.

Consequently, the small variance in BERTSCORE-F1 does not necessarily indicate equivalent personalization quality. Rather, it reflects the metric’s limitation in capturing structural or stylistic personalization effects. Therefore, the proposed Persona Hit Score (PHS) serves as a more suitable metric for evaluating nuanced user preference reflection and individualized alignment in text generation.

In Experiment 2, we observed a slight improvement in accuracy as the batch size \mathcal{D}_{batch} increased. For Continuous Alignment, frequent updates over very short dialog spans tended to cause overfitting to transient user preferences. While the optimal batch size may depend on the specific use case, updating the Preference Description based on moderately sized dialog batches appears to better capture consistent user tendencies and preferences. Furthermore, we found that non Reasoning models were more effective in representing user preferences than Reasoning models, likely because the dataset primarily consists of casual, daily conversations. Future work will expand the dataset to include more complex, reasoning oriented dialogs to further examine the effectiveness of the proposed method under such conditions.

It was also confirmed that the use of RAG (Retrieval Augmented Generation) plays an important role in continuous preference updating. By utilizing RAG, it becomes possible to retrieve relevant context from past Episodic Memory and integrate it with the current dialog content, allowing the system to reflect changes in user preferences without inconsistencies. In fact, under the RAG condition, accuracy remained consistently higher than under the non RAG condition, suggesting that response generation was better aligned with users’ historical tendencies. These findings indicate that updating preference representations does not rely solely on recent dialogs but requires a mechanism that dynamically references past dialog logs.

On the other hand, the evaluation of the PersonaMem dataset was conducted based on classification accuracy under the assumption of a single correct answer, which may not fully capture the effects of RAG, such as partial reflection of preferences or stylistic consistency. In the future, it will be necessary to introduce ranking based metrics and human evaluations to more precisely assess qualitative improvements in preference updating via RAG. Furthermore, dynamically controlling the retrieval scope and update frequency of RAG is an important direction for future work, as it could help suppress overfitting to short-term preference changes while maintaining long-term consistency.

Conclusion

We presented a dynamic personalization framework that models each user preference as a vector with polarity and intensity, aggregates past and current signals through arithmetic composition, and updates the representation at test time via a lightweight procedure. This design reconciles short-term tendencies with long-term memory and reduces contradictions in the personalized prompt.

On ImplicitPersona datasets, our method consistently improves Persona Hit Score (PHS) over a non-personalized baseline and a strong prompt-engineering baseline across multiple LLMs—yielding +2–8 absolute points depending on the model/judge configuration. On PersonaMem datasets, continuous preference updates enhance answer selection accuracy in long-turn dialogs, while ablations indicate that intensity/polarity granularity may be task-dependent.

Our study has limitations: evaluations largely rely on LLM-as-a-Judge and simulation datasets; statistical significance and human studies are left for future work. We also observed cases where coarse preference descriptors are competitive, motivating deeper analysis of representation granularity and update rules.

In future work, we plan to (i) expand beyond QA to agentic applications (e.g., planning and recommendation) with online user studies, (ii) strengthen benchmarks and metrics (calibrated judges, inter-judge agreement), (iii) explore privacy-preserving and efficient deployment for many users, and (iv) investigate principled decay/regularization for lifelong personalization. We hope these results contribute to PerFM’s goals on memory-based personalization, robust evaluation, and trustworthy deployment.

Acknowledgments

I gratefully acknowledge the support and assistance I received from many individuals, both directly and indirectly, throughout the course of this research. I extend my deepest appreciation to all those whose contributions, in any form, helped bring this work to fruition.

References

- Anthropic. 2024. Claude Introduces Memory for Teams at Work. <https://www.anthropic.com/news/memory>. Accessed: October 2, 2025.
- Banerjee, S.; and Lavie, A. 2005. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In Goldstein, J.; Lavie, A.; Lin, C.-Y.; and Voss, C., eds., *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, 65–72. Ann Arbor, Michigan: Association for Computational Linguistics.
- Chang, Y.; Wang, X.; Wang, J.; Wu, Y.; Yang, L.; Zhu, K.; Chen, H.; Yi, X.; Wang, C.; Wang, Y.; Ye, W.; Zhang, Y.; Chang, Y.; Yu, P. S.; Yang, Q.; and Xie, X. 2024. A Survey on Evaluation of Large Language Models. *ACM Trans. Intell. Syst. Technol.*, 15(3).
- Dinan, E.; Logacheva, V.; Malykh, V.; Miller, A.; Shuster, K.; Urbanek, J.; Kiela, D.; Szlam, A.; Serban, I.; Lowe,

- R.; Prabhumoye, S.; Black, A. W.; Rudnicky, A.; Williams, J.; Pineau, J.; Burtsev, M.; and Weston, J. 2019. The Second Conversational Intelligence Challenge (ConvAI2). arXiv:1902.00098.
- Gu, J.; Jiang, X.; Shi, Z.; Tan, H.; Zhai, X.; Xu, C.; Li, W.; Shen, Y.; Ma, S.; Liu, H.; Wang, S.; Zhang, K.; Wang, Y.; Gao, W.; Ni, L.; and Guo, J. 2025. A Survey on LLM-as-a-Judge. arXiv:2411.15594.
- Hutto, C.; and Gilbert, E. 2014. VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. *Proceedings of the International AAAI Conference on Web and Social Media*, 8(1): 216–225.
- Jang, J.; Kim, S.; Lin, B. Y.; Wang, Y.; Hessel, J.; Zettlemoyer, L.; Hajishirzi, H.; Choi, Y.; and Ammanabrolu, P. 2023. Personalized Soups: Personalized Large Language Model Alignment via Post-hoc Parameter Merging. arXiv:2310.11564.
- Jiang, B.; Hao, Z.; Cho, Y.-M.; Li, B.; Yuan, Y.; Chen, S.; Ungar, L.; Taylor, C. J.; and Roth, D. 2025. Know Me, Respond to Me: Benchmarking LLMs for Dynamic User Profiling and Personalized Responses at Scale. *arXiv preprint arXiv:2504.14225*.
- Kirk, H. R.; Whitefield, A.; Röttger, P.; Bean, A.; Margatina, K.; Ciro, J.; Mosquera, R.; Bartolo, M.; Williams, A.; He, H.; Vidgen, B.; and Hale, S. A. 2024. The PRISM Alignment Dataset: What Participatory, Representative and Individualised Human Feedback Reveals About the Subjective and Multicultural Alignment of Large Language Models. arXiv:2404.16019.
- Leong, J.; Pataranutaporn, P.; Danry, V.; Perteneder, F.; Mao, Y.; and Maes, P. 2024. Putting Things into Context: Generative AI-Enabled Context Personalization for Vocabulary Learning Improves Learning Motivation. In *Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI '24)*.
- Lin, C.-Y. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*, 74–81. Barcelona, Spain: Association for Computational Linguistics.
- Liu, J.; Qiu, Z.; Li, Z.; Dai, Q.; Yu, W.; Zhu, J.; Hu, M.; Yang, M.; Chua, T.-S.; and King, I. 2025. A Survey of Personalized Large Language Models: Progress and Future Directions. arXiv:2502.11528.
- Ni, J.; Li, J.; and McAuley, J. 2019. Justifying Recommendations using Distantly-Labeled Reviews and Fine-Grained Aspects. In Inui, K.; Jiang, J.; Ng, V.; and Wan, X., eds., *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 188–197. Hong Kong, China: Association for Computational Linguistics.
- OpenAI. 2024. GPT-4o System Card. Technical report, OpenAI. Available at <https://cdn.openai.com/gpt-4o-system-card.pdf>.
- OpenAI. 2025a. GPT-5 System Card. Technical report, OpenAI. Available at <https://cdn.openai.com/gpt-5-system-card.pdf>.
- OpenAI. 2025b. Memory and new controls for ChatGPT. <https://openai.com/ja-JP/index/memory-and-new-controls-for-chatgpt/>. Accessed: October 2, 2025.
- Salemi, A.; Mysore, S.; Bendersky, M.; and Zamani, H. 2024. LaMP: When Large Language Models Meet Personalization. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL 2024)*.
- Samarati, P.; and Sweeney, L. 1998. Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression.
- Sanner, S.; Balog, K.; Radlinski, F.; Wedin, B.; and Dixon, L. 2023. Large Language Models are Competitive Near Cold-start Recommenders for Language- and Item-based Preferences. In *Proceedings of the 17th ACM Conference on Recommender Systems (RecSys '23)*.
- Tulving, E. 1972. Episodic and semantic memory.
- Wang, W.; Dong, L.; Cheng, H.; Liu, X.; Yan, X.; Gao, J.; and Wei, F. 2023. Augmenting Language Models with Long-Term Memory. In *Advances in Neural Information Processing Systems (NeurIPS 2023)*.
- Zhang, W.; Zhang, X.; Zhang, C.; Yang, L.; Shang, J.; Wei, Z.; Zou, H. P.; Huang, Z.; Wang, Z.; Gao, Y.; Pan, X.; Xiong, L.; Liu, J.; Yu, P. S.; and Li, X. 2025a. PersonaAgent: When Large Language Model Agents Meet Personalization at Test Time. arXiv:2506.06254.
- Zhang, Z.; Rossi, R. A.; Kveton, B.; Shao, Y.; Yang, D.; Zamani, H.; Derroncourt, F.; Barrow, J.; Yu, T.; Kim, S.; Zhang, R.; Gu, J.; Derr, T.; Chen, H.; Wu, J.; Chen, X.; Wang, Z.; Mitra, S.; Lipka, N.; Ahmed, N. K.; and Wang, Y. 2025b. Personalization of Large Language Models: A Survey. *Transactions on Machine Learning Research*. Survey Certification.
- Zheng, L.; Chiang, W.-L.; Sheng, Y.; Zhuang, S.; Wu, Z.; Zhuang, Y.; Lin, Z.; Li, Z.; Li, D.; Xing, E. P.; Zhang, H.; Gonzalez, J. E.; and Stoica, I. 2023. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. arXiv:2306.05685.

Appendix

A. Prompt Initialization with Preference Description

You are a helpful, personalized assistant. Answer the user's questions while taking into account the user's information and preferences.

Description of user preferences

[Preference Description]

Explanation of the Preference Description

- **text:** The content of the user's preference
- **polarity:** The polarity of the preference (1: likes, 0: neutral, -1: dislikes)
- **intensity:** The strength of the preference (0: indifferent, 5: very strong)

Warnings for Tool Use

- Think step by step about the information you need.
- Provide clear and concise answers. Do not include explanations in the final answer.

B. Preference Description Update

Instruction Based on the given user's current **Preference Description** and the provided **Candidate Preference Description**, update the user's preference to better reflect their attributes and tastes. Polarity and intensity should be treated as pseudo arithmetic values — when a positive and a negative polarity cancel each other out, adjust the intensity accordingly.

Information

- **Current Preference Description:**
[preference_description]
- **Candidate Preference Description:**
[subpreference_description]

Additional Notes

- **polarity:** -1 = negative, 0 = neutral, 1 = positive
- **intensity:** ranges from 0 to 5 (the higher the number, the stronger the preference)

Guidelines for Updating

- If **polarity = 0**, then set **intensity = 0**.
- Update the Preference Description based on the candidate Preference Description as needed.
- Keep preferences that are not mentioned in the candidate description unchanged.
- When updating based on the candidate description, appropriately adjust **polarity** and **intensity** according to their combined effects.

C. Persona Hit Score(PHS)

Evaluate the consistency between the given [**Persona (Preference Description)**] and [**Generated Text**] according to the criteria below.

When the persona's nuance (intensity and polarity) is well reflected, please take that into account in the scoring.

Scoring Criteria

- **Relevance (0–25)**
- **Coverage (0–25)**
- **Specificity (0–25)**
- **Evidence/Consistency (0–25)**

The total score should be between **0** and **100**.

Persona (Preference Description)

[Preference Description]

Explanation of the Preference Description

- **text:** The content of the user's preference
- **polarity:** The polarity of the preference (1: likes, 0: neutral, -1: dislikes)
- **intensity:** The strength of the preference (0: indifferent, 5: very strong)

Generated Text

[generated text]

Notes for Evaluation

- Deduct points for *assertive statements* not explicitly stated in the persona.
- Rewordings or differences between English and Japanese expressions are acceptable, but *forced interpretations* are not.
- Add points for specific actions, proper nouns, numbers, or procedures.

D.Tool for Retrieve Episodic Memory Prompt

Retrieves the top-5 most relevant items or histories from the user memory using RAG (Retrieval Augmented Generation).

Suitable Use Cases

- Search for detailed information about related items
- Answer specific questions based on personal data
- Incorporate user preferences into the final response

Input: A specific search query or question related to the content

Output: Related dialog histories retrieved from the user memory

Notes: The more specific the query, the more accurate the results will be.

Requirements: When answering a question, this tool must be used at least once. Queries should be generated in English.