Preference-Aware Memory Update for Long-Term LLM Agents

Haoran Sun[1,2*], **Zekun Zhang**[2*], **Shaoning Zeng**[1†]
[1] Yangtze Delta Region Institute (Huzhou), University of Electronic Science and Technology of China, Huzhou, China
[2] School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu, China

2022090916002@std.uestc.edu.cn, 2023090909020@std.uestc.edu.cn, zsn@outlook.com

Abstract

One of the key factors influencing the reasoning capabilities of LLM-based agents is their ability to leverage long-term memory. Integrating long-term memory mechanisms allows agents to make informed decisions grounded in historical interactions. While recent advances have significantly improved the storage and retrieval components—e.g., by encoding memory into dense vectors for similarity search or organizing memory as structured knowledge graphs—most existing approaches fall short in memory updating. In particular, they lack mechanisms for dynamically refining preference memory representations in response to evolving user behaviors and contexts. To address this gap, we propose a Preference-Aware Memory Update Mechanism (PAMU) that enables dynamic and personalized memory refinement. By integrating sliding window averages (SW) with exponential moving averages (EMA), PAMU constructs a fused preference-aware representation that captures both short-term fluctuations and long-term user tendencies. We conduct experiments on five task scenarios of the LoCoMo dataset, and the results show that our mechanism can significantly improve the output quality of LLM in five baselines, validating its effectiveness in long-term conversations.

1 Introduction

Large Language Model (LLM) agents exhibit strong autonomous decision-making capabilities across a wide range of tasks, particularly excelling in open-domain question answering (Yao et al., 2024; Huang et al., 2024a; DeepSeek-AI, 2025). In long-term dialogue scenarios, effective reasoning and decision-making often require integrating past interactions, making internal memory mechanisms essential (Zhang et al., 2024a,b). These mechanisms aim to emulate human-like cognitive memory by retaining prior conversational context, enabling the agent to retrieve relevant information and generate context-aware, personalized responses. The design and adaptation of such memory systems are thus critical to the agent's performance in complex, temporally extended tasks (Li et al., 2024; Guo et al., 2024; Sun et al., 2025; Sun and Zeng, 2025).

The most basic memory approach concatenates prior dialogues with the current prompt, but this method is constrained by the LLM's finite context window, limiting its effectiveness in prolonged interactions (Jin et al., 2024; Gu et al., 2024). To address this, recent studies have explored more sophisticated architectures: MemoryBank (Zhong et al., 2024) encodes past information into dense vectors and retrieves memories via similarity search ; MemGPT (Packer et al., 2023) introduces a hierarchical OS-inspired memory system that combines limited-context attention with external memory storage, yet suffers from a trade-off between retrieval accuracy and efficiency; MemInsight (Salama et al., 2025) enhances memory representation by autonomously extracting structured key-value attributes; and A-MEM (Xu et al., 2025; Sun et al., 2026), inspired by the Zettelkasten method, dynamically constructs evolving knowledge graphs for self-organizing memory.

Despite these advances, existing systems predominantly focus on memory storage and retrieval, often overlooking a crucial aspect: how to adaptively and continuously update memory in response to evolving user behavior during long-term interactions (Wu et al., 2025; Huang et al., 2024b). In real-world deployment, users are non-stationary—their intents, preferences, and goals shift over time. Without dynamic memory updating, agents risk relying on outdated or misaligned information, leading to degraded performance and user trust.

To bridge this gap, we propose a Preference-

[*] These authors contributed equally to this work.
[†] Corresponding author.

Findings of the Association for Computational Linguistics: ACL 2026, pages 783–793
July 2-7, 2026

Aware Memory Update Mechanism that enables LLMs to perceive, adapt to, and respond in alignment with evolving user preferences. At its core is a novel Preference Change Perception Module, which combines a sliding window average and an exponential moving average (EMA) to construct a dual-perspective user preference representation—capturing short-term behavioral shifts while robustly modeling long-term trends. We further introduce a formalized change detection signal, triggered by the deviation between short- and long-term estimates, to guide when and how memory updates should occur. This allows for interpretable and controllable adaptation in response to preference drift. Notably, our mechanism is highly modular and model-agnostic: it requires no fine-tuning or architectural modification and can be seamlessly integrated into existing memory-augmented LLM frameworks.

2 Related Work

To enhance the long-term reasoning capabilities of LLM agents, various memory systems have been proposed. ReadAgent (Lee et al., 2024) segments and compresses documents into key-point memories for retrieval-augmented reading comprehension. MemGPT (Packer et al., 2023) uses OS-inspired virtual memory management, combining hierarchical memory with external storage via dynamic function calls. SCM (Wang et al., 2023) enables agents to autonomously decide when and how to access memory through a controller-stream-agent framework. MemoryBank (Zhong et al., 2024), grounded in the Ebbinghaus forgetting curve, supports memory storage, retrieval, and update for user-aware personalization. A-MEM (Xu et al., 2025), inspired by Zettelkasten, organizes memory as evolving, self-linked knowledge notes. MemInsight enhances memory representation by extracting structured attributes for more accurate semantic retrieval (Salama et al., 2025).

While these approaches have advanced memory modeling in LLMs—especially in storage, retrieval, and organization—they largely assume static user behavior. In practice, user preferences and goals evolve dynamically. However, existing systems lack mechanisms to adaptively track and update memory in response to such changes. This highlights a critical gap: the need for a dynamic, preference-aware memory update mechanism that supports long-term personalization in LLM agents.

3 Methodology

In this section, we introduce our Preference-Aware Memory Update (PAMU) mechanism.

3.1 Preference Extractor

The system constructs a user preference vector $\mathbf{P} = \{p_1, p_2, \ldots, p_D\}$ by extracting multidimensional preference signals from multi-turn interactions between the user and the model. Each dimension p_d represents a specific user preference type, such as tone style, response length, emotional tone, information density, and degree of formality. After each dialogue turn, the system updates the preference vector by analyzing user feedback and linguistic features. Specifically:

- **Tone Style.** A RoBERTa encoder with a multi-class classification head is employed to analyze the stylistic features of user utterances. The model produces a probability distribution over predefined tone categories. The category with the highest probability and its score are concatenated into a tuple to represent the tone dimension.
- **Response Length.** This is measured by the number of tokens generated by the model. The average response length over the past K turns is computed and normalized to the [0, 1] range to form the length dimension.
- **Emotional Tone.** An emotion classification model identifies the dominant emotional categories from both user and assistant utterances. A probability vector over predefined emotional classes is extracted, and the class with the highest probability is used, along with its score, to represent the emotional tone dimension.
- **Information Density.** The system leverages an OpenIE model to extract structured (subject, predicate, object) triples from the assistant's responses. Each triple is treated as an atomic information unit. The number of extracted triples per turn is treated as the count of knowledge points. The information density ID_t of the response at turn t is defined as:

$$ID_t = \frac{K_t}{L_t} \tag{1}$$

Among them, K_t represents the number of triples sampled in the t-th turn, and L_t represents the total number of words in the response

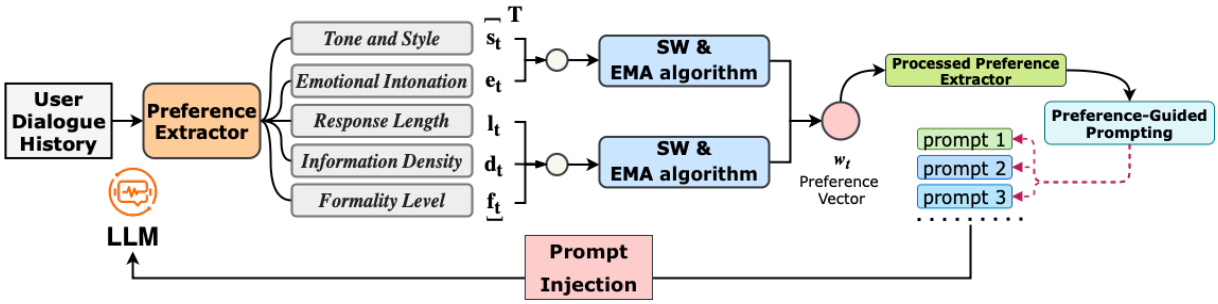

Figure 1: **Illustration of PAMU method.** PAMU extracts user preferences from dialogue, models short- and long-term trends via SW and EMA, detects preference shifts, and updates the prompt to guide personalized generation.

of that turn. This ratio measures the average amount of information carried by each word, reflecting the compactness of language use and the degree of knowledge density.

- **Degree of Formality.** A pretrained formality classification model is employed to evaluate the assistant's response, yielding a normalized formality score within the range [0, 1], where 0 indicates fully colloquial (spoken) language and 1 denotes fully formal (written) language. This score is directly used as the value for the formality dimension.

Accordingly, for each dialogue turn, the system extracts a five-dimensional user preference vector:

$$\mathbf{p}_t = (s_t, l_t, e_t, d_t, f_t) \tag{2}$$

Here, s_t and e_t denote tuples containing the predicted category index and its probability for tone style and emotional tone, respectively; l_t, d_t, and f_t are normalized scalar values representing response length, information density, and formality. This vector is then fed into the Preference Shift Detector, which models the temporal dynamics of user preferences using a combination of a sliding window mechanism and Exponential Moving Average (EMA). This enables the system to detect both gradual drifts and abrupt shifts in preferences, and to determine whether the model's response strategy requires adaptation or fine-tuning to better align with evolving user intent.

3.2 Preference Change Perception Mechanism

Following the extraction of multi-dimensional user preference vectors, a Preference Dynamics Perception Module is employed to model behavioral shifts and enable personalized response adaptation. This module integrates Sliding Window (SW) averaging with Exponential Moving Average (EMA) to continuously update preference estimates at each dialogue turn, thereby guiding the response generator toward controlled, user-aligned outputs.

To provide rigorous theoretical justification for this design, we ground the SW+EMA fusion in classical time-series analysis and HCI user modeling literature. Sliding Window averaging is highly sensitive to recent behavioral shifts, making it ideal for capturing abrupt preference changes in non-stationary interactions. In contrast, EMA applies exponential decay to emphasize long-term trends while suppressing transient noise, a technique proven effective in adaptive recommendation and user modeling systems. The weighted fusion

$$\hat{w}_t^{(d)} = \lambda \cdot SW_t^{(d)} + (1-\lambda) \cdot EMA_t^{(d)}$$

(Eqs. (6), (9), (12)) explicitly balances responsiveness (λ) and stability. To handle potential conflicts between short- and long-term signals (e.g., contradictory preferences), we further introduce a divergence-based change detection signal $C_t^{(d)} = |SW_t^{(d)} - EMA_t^{(d)}|$, which triggers prompt/memory updates only when the deviation exceeds a threshold δ. This mechanism ensures interpretability, prevents over-reaction to noise, and maintains robustness across different model scales and conversation lengths. Hyperparameter sensitivity is validated in our ablation studies (Section 5), confirming stable performance for moderate values of $\lambda \in [0.4, 0.7]$ and $\beta \in [0.8, 0.95]$.

Specifically, we uniformly represent user preference vectors in the form of:

$$\boldsymbol{p}_t = \left[p_t^{(1)}, p_t^{(2)}, \ldots, p_t^{(D)}\right] \tag{3}$$

Among them, D represents the number of preference dimensions. Each dimensional preference value $p_t^{(d)}$ may be a continuous variable (such as response length, information density, formality level)

or a categorical variable (such as tone style, emotional intonation). For categorical variables, we use the tuple $(c_t^{(d)}, q_t^{(d)})$ to represent, where $c_t^{(d)}$ is the category index and $q_t^{(d)}$ is the categorical probability distribution.

3.2.1 Dynamic Modeling of Continuous Preference Dimensions.

For continuous preference dimensions (length, information density, and Degree of formalization), we define a sliding window of length W to calculate the sliding average preference value at the current time t.

$$\mathrm{SW}_t^{(d)} = \frac{1}{W} \sum_{i=t-W+1}^{t} p_i^{(d)} \tag{4}$$

Among them, $\mathrm{SW}_t^{(d)}$ is the sliding window average of the preference in the d-th dimension at time t; W is the sliding window length (the number of historical turns used to calculate the average); $p_i^{(d)}$ represents the preference value in the d-th dimension of the i-th turn; $\sum_{i=t-W+1}^{t}$ denotes the cumulative operation on the preference values from the $(t-W+1)$-th turn to the t-th turn within the window.

Meanwhile, Exponential Moving Average (EMA) is introduced to enhance the memory capacity for long-term trends. Let EMAt(d) denote the exponential average of preference dimension d at time t, then its update formula is:

$$\mathrm{EMA}_t^{(d)} = \beta \cdot \mathrm{EMA}_{t-1}^{(d)} + (1-\beta) \cdot p_t^{(d)} \tag{5}$$

Among them, $\beta \in (0,1)$ is the decay coefficient, which controls the degree of influence of historical preferences on the current estimate. SW is more sensitive to recent preference changes, while EMA is used to slowly track long-term trends.

After the combination of the two, the fused perception vector is defined as:

$$\hat{w}_t^{(d)} = \lambda \cdot \mathrm{SW}_t^{(d)} + (1-\lambda) \cdot \mathrm{EMA}_t^{(d)} \tag{6}$$

Among them, $\lambda \in [0,1]$ controls the weight proportion of the sliding window and exponential average. This fusion strategy can flexibly adapt to the fast-changing and slow-changing characteristics in user preferences.

3.2.2 Dynamic Modeling of Categorical Preference Dimensions.

For categorical dimensions (tone style and emotional intonation), we represent the preference of each turn as $(c_t^{(d)}, q_t^{(d)})$, which is the currently most likely category and its corresponding probability distribution. We perform sliding average and exponential average on the category probability distribution vectors respectively:

$$\mathrm{SW}_t^{(d)} = \frac{1}{W} \sum_{i=t-W+1}^{t} q_i^{(d)} \tag{7}$$

$$\mathrm{EMA}_t^{(d)} = \beta \cdot \mathrm{EMA}_{t-1}^{(d)} + (1-\beta) \cdot q_t^{(d)} \tag{8}$$

The fused category probability perception vector is:

$$\hat{w}_t^{(d)} = \lambda \cdot \mathrm{SW}_t^{(d)} + (1-\lambda) \cdot \mathrm{EMA}_t^{(d)} \tag{9}$$

Select the category with the highest probability as the control label to be used during generation at the current time:

$$c_t^{(d)} = \arg\max_j \hat{w}_t^{(d)}[j], \tag{10}$$

where j is the category index.

3.3 Preference-Guided Prompting

To enable personalized generation, we explicitly inject the fused user preference vector $\mathbf{w}_t$ into a structured natural language prompt. This guides the LLM to produce outputs aligned with the user's desired style and attributes, without modifying the model architecture or decoder—achieving flexible behavior control purely via prompt engineering.

Compared to fine-tuning-based implicit modeling, this approach is more efficient, interpretable, and adaptable at inference time, avoiding issues like catastrophic forgetting and supporting real-time preference updates in multi-user or multi-domain settings.

Concretely, $\mathbf{w}_t$ is converted into a textual instruction embedded in the prompt, e.g., "Please answer the following question in the style of: [Tone: humorous], [Emotion: relaxed], [Information density: moderate], [Length: brief]."

Each preference in the prompt is derived from the current dialogue turn, using a fusion of sliding window averaging and exponential moving average (EMA) to smooth short-term fluctuations. The formatting of different preference types is as follows:

Algorithm 1 Preference-Aware Generation

procedure GENERATERESPONSE(H_t, x_t) ▷ History and current user input
  $p_t \leftarrow$ EXTRACTPREFERENCES(H_t, x_t) ▷ $p_t = (s_t, l_t, e_t, d_t, f_t)$
  for all $d \in$ {tone, length, emotion, density, formality} **do**
    $SW_t[d] \leftarrow$ Mean($p_{t-W+1:t}[d]$) ▷ Sliding window average
    $EMA_t[d] \leftarrow \beta \cdot EMA_{t-1}[d] + (1-\beta) \cdot p_t[d]$
    $w_t[d] \leftarrow \lambda \cdot SW_t[d] + (1-\lambda) \cdot EMA_t[d]$
  end for
  $desc \leftarrow$ FORMATPREFERENCE(w_t) ▷ Natural language preference prompt
  $prompt \leftarrow$ `"Respond in style: "` $+ desc +$ `"\n"` $+ x_t$
  $y_t \leftarrow$ LLM.generate($prompt$)
end procedure
function EXTRACTPREFERENCES(H_t, x_t)
  $s_t \leftarrow$ ToneClassifier(x_t, H_t) ▷ Categorical: RoBERTa-based
  $l_t \leftarrow$ Normalize(MeanLength($r_{t-K:t-1}$))
  $e_t \leftarrow$ EmotionAnalyzer(x_t, H_t)
  $d_t \leftarrow$ InfoDensity(r_{t-1}) ▷ Triple/token ratio
  $f_t \leftarrow$ FormalityDetector(x_t)
  return $(s_t, l_t, e_t, d_t, f_t)$
end function

- **Categorical dimensions** (e.g., tone style, emotional tone) represented as tuples (c, p), where c is the index of the most probable category and p its confidence score. The selected label c^* is verbalized into descriptors such as "humorous," "serious," or "gentle" for prompt inclusion.

- **Continuous dimensions** (e.g., response length, information density, formality) maintained as scalar values, discretized into predefined intervals and mapped to interpretable semantic tags (e.g., "brief," "detailed") to enhance the model's understanding of intensity and alignment strength.

For the information density value $d \in [0, 1]$, we define a discretization function that maps continuous preference scores into interpretable semantic tags:

$$\text{Label}(d) = \begin{cases} \text{Sparse}, & d \in [0, 0.33) \\ \text{Moderate}, & d \in [0.33, 0.66) \\ \text{Dense}, & d \in [0.66, 1] \end{cases} \quad (11)$$

This mapping strategy is applied uniformly to all continuous preference dimensions (e.g., response length, information density, formality). By concatenating the resulting descriptors across dimensions, a complete structured control prompt can be automatically constructed. Such an explicit prompting mechanism enables the preference vector to function not only as a soft controller for generation, but also as an interpretable interface for user-aligned output control. Owing to its model-agnostic nature, this mechanism is highly extensible and applicable to a wide range of downstream tasks, including multi-turn dialogue generation, personalized question answering, and preference-aware memory systems.

3.4 Motivation and Basis

In long-term human-computer interaction scenarios, user behavior exhibits strong non-stationarity. Users' tone styles, emotional states, information density requirements, and degrees of formality often undergo gradual evolution or abrupt changes due to factors such as task context, personal emotions, and interaction stages. Although existing memory mechanisms have made progress in information storage and retrieval, they generally rely on a core assumption: that user preferences are stable or uniformly distributed over time. This static assumption may lead the model to generate responses based on outdated preferences, reducing dialogue consistency and user satisfaction. Therefore, our memory update mechanism must possess sensitivity and behavioral interpretability.

In time-series modeling, Sliding Window Average and Exponential Moving Average (EMA) are two commonly used but complementary techniques. Sliding Window Average is sensitive to recent changes and is suitable for capturing short-term preference fluctuations, while EMA focuses on long-term trends through exponential decay, filtering out local noise and modeling inertial behavior. Thus, we propose to integrate the two, constructing a preference perception vector that is both responsive and stable, allowing the model to balance its response style between short-term personalization and long-term consistency:

$$\hat{w}_t^{(d)} = \lambda \cdot \mathrm{SW}_t^{(d)} + (1-\lambda) \cdot \mathrm{EMA}_t^{(d)} \quad (12)$$

Where, $\mathrm{SW}_t^{(d)} = \frac{1}{W}\sum_{i=t-W+1}^{t} p_i^{(d)}$ represents the recent average of preferences; $\mathrm{EMA}_t^{(d)} = \beta \cdot \mathrm{EMA}_{t-1}^{(d)} + (1-\beta) \cdot p_t^{(d)}$ represents the smoothed trend of historical preferences. $\lambda \in [0,1]$ controls the degree of attention to short-term changes, and $\beta \in (0,1)$ controls the memory depth of long-term trends.

This mechanism addresses the core problem proposed in this paper: how to dynamically update user preference memory within LLM agents and accordingly adjust their responses in real time.

Handling Preference Conflicts and Cold-Start. When short- and long-term signals conflict (i.e., $C_t^{(d)} > \delta$), the change detection signal prioritizes the short-term SW estimate, reflecting the user's most recent intent. For cold-start scenarios (first 5 turns), we initialize EMA with the global average from the first W turns and set $\lambda = 1.0$ (full SW weight) until sufficient history is accumulated. This design ensures graceful degradation and rapid adaptation even in new conversations or with sparse signals.

4 Experiment

4.1 Setup

Dataset and Evaluation Metrics. To evaluate whether our preference update mechanism can effectively guide LLMs to generate user-aligned responses in long-term multi-turn dialogue scenarios, we adopt the LoCoMo dataset (Maharana et al., 2024) following previous related work (Xu et al., 2025; Zhong et al., 2024). LoCoMo is specifically designed to assess the memory and consistency capabilities of LLM-based agents in extended multi-session interactions. Key characteristics of the dataset include 50 dialogues, each with an average of 300 turns, spanning up to 35 distinct sessions and approximately 9,000 tokens per dialogue. We choose three types of task in it:

- **Single-hop questions (SH.)**: answerable within a single session (2,705 pairs).
- **Multi-hop questions (MH.)**: requiring cross-session information aggregation (1,104 pairs).
- **Temporal reasoning (T.)**: testing understanding of time-sensitive information (1,547 pairs).

LoCoMo emphasizes long-range contextual coherence across sessions, making it a robust benchmark for evaluating LLMs' ability to handle memory-dependent reasoning and maintain response consistency in long-term interactions. We employ two primary metrics to comprehensively assess model performance under different memory settings: F1 Score and BLUE-1 Score. These metrics jointly assess the effectiveness of our mechanism in enhancing user-aligned generation in long-context conversational settings.

Baselines. As our work specifically focuses on preference memory update mechanisms rather than proposing a complete memory framework, we evaluate the effectiveness of our approach by integrating it into five representative long-term memory methods and conducting before-and-after comparisons. The selected baselines include: **ReadAgent (RA.)** (Lee et al., 2024), **MemoryBank (MB.)** (Zhong et al., 2024), **MemGPT (MG.)** (Packer et al., 2023), and **A-MEM (AM.)** (Xu et al., 2025), all of which are currently very mainstream memory frameworks. For each method, we augment its original architecture by appending our preference update module, without modifying its internal memory operations or update logic. Importantly, our mechanism is fully compatible and modular, operating independently of each baseline's native update strategy. The only difference between the original and enhanced versions lies in the presence of our preference-aware update component, ensuring that any observed performance gains can be attributed solely to our proposed mechanism.

Implementation Details. In our experiments, we utilize three families of large language models with varying scales—Qwen 2.5-1.5B / 3B (Yang et al., 2024), LLaMA-7B / 30B (Touvron et al., 2023), and LLaMA 3.2-1.5B / 3B (Touvron et al., 2023)—as the base QA models. These diverse model types and sizes allow for a more comprehensive evaluation of the robustness and generalizability of our proposed mechanism. All models are deployed locally via Ollama. For our preference signal extraction, we employ the following pretrained models for each corresponding dimension: RoBERTa encoder with a multi-class classification head (Tone Style); Open-source pretrained SKEP (Tian et al., 2020) model (Emotional Tone); Knowledge tuples extracted via OpenNRE (Han

et al., 2019), representing structured semantic units (Information Density). For fair comparison and reproducibility, we plug the same preference module into every baseline memory system without altering their original architectures or reasoning logic, performing only the minimal interface adjustments needed to ingest the preference vector. At inference, each model is fed only the input question and its own historical memory; the final preference prompt, computed from the vector, is simply appended to the original prompt of each method as an explicit control signal for generation.

4.2 Main Results and Analysis

Each result represents the average over three independent runs with different random seeds. We conducted paired t-tests among baselines. Results marked with * indicate statistically significant improvements ($p < 0.05$). † indicates the model is equipped with our proposed Preference-Aware Memory Update (PAMU) mechanism. The format of all results is **Before Augment / After Augment**.

Comparison Analysis. Table 2, 1 and 3 report results on three representative tasks. On both single-hop and multi-hop reasoning, every baseline upgraded with PAMU shows large gains in response quality while keeping or slightly raising accuracy, demonstrating that PAMU improves generation without hurting correctness. Most strikingly, on temporal reasoning PAMU boosts both accuracy and quality by a wide margin, proving it can spot short-term preference shifts and simultaneously update long-term user trends.

Methods	Single-Hop	
	F1	BLUE-1
RA. / RA.†	6.54 / **8.27**	4.87 / **8.97***
MB. / MB.†	11.14 / **12.34**	8.24 / **10.57***
MG. / MG.†	10.43 / **10.49**	7.54 / **11.46***
AM. / AM.†	17.24 / **17.93**	11.35 / **15.73***
RA. / RA.†	3.23 / 3.23	2.89 / **4.23***
MB. / MB.†	3.54 / **3.87**	3.39 / **7.35***
MG. / MG.†	5.07 / **5.24**	4.28 / **8.65***
AM. / AM.†	12.52 / **13.23**	9.24 / **13.24***

Table 1: Experimental results on single-hop tasks using Qwen 2.5-1.5B (upper part)/3B (lower part) models.

Ablation Study. To systematically evaluate the individual contributions of each component in our proposed preference-aware memory update mecha-

Methods	Multi-Hop	
	F1	BLUE-1
RA. / RA.†	2.45 / **2.98**	2.67 / **5.34***
MB. / MB.†	7.61 / **6.03**	6.56 / **9.23***
MG. / MG.†	5.23 / **6.78**	5.14 / **10.87***
AM. / AM.†	16.57 / **17.02**	11.24 / **19.23***
RA. / RA.†	3.05 / **3.67**	2.67 / **5.45***
MB. / MB.†	3.56 / 3.56	3.02 / **7.65***
MG. / MG.†	3.02 / 3.02	2.95 / **6.34***
AM. / AM.†	19.35 / **20.14**	13.27 / **23.14***

Table 2: Experimental results on multi-hop tasks using LLaMA 3.2-1.5B (upper part)/3B (lower part) models.

Methods	Temporal Reasoning	
	F1	BLUE-1
RA. / RA.†	12.24 / **15.45***	11.17 / **15.67***
MB. / MB.†	14.56 / **19.76***	11.95 / **17.24***
MG. / MG.†	11.14 / **17.54***	8.24 / **15.57***
AM. / AM.†	17.55 / **23.23***	14.67 / **21.46***
RA. / RA.†	5.57 / **7.67***	5.22 / **7.43***
MB. / MB.†	4.77 / **8.98***	4.87 / **7.34***
MG. / MG.†	5.64 / **9.95***	5.53 / **8.24***
AM. / AM.†	12.54 / **19.87***	11.85 / **18.23***

Table 3: Experimental results on temporal reasoning tasks using LLaMA-7B (upper part)/30B (lower part) models.

nism, we conduct a comprehensive set of ablation studies. The details of each ablation and its corresponding replacement are as follows:

Sliding Window Average (w/o SW): ablation removes SW so only EMA remains, simulating loss of short-term responsiveness. **Exponential Moving Average (w/o EMA):** Removing EMA isolates the effect of losing long-term stability. **Fusion Mechanism (Equal Fusion):** Ablation fixes $\lambda = 0.5$, disabling adaptive balancing. **Preference Change Detection (w/o Detection):** Removes the divergence-based change signal, preventing prompt/memory adaptation and reverting to static generation templates. **Prompt Injection (w/o Prompt):** Eliminates explicit preference prompts, providing only raw user input to test generation without direct conditioning. **Multi-Dimensional Preference Modeling (Single Pref):** Reduces the 5D preference vector (tone, length, emotion, density, formality) to a single feature (e.g., length) to assess the benefit of multi-dimensional modeling. **Dynamic vs. Static Preference Modeling (Static Pref):** Re-

Turn	Tone.	Length	Emotion	Density	Formality
1	(Humor, 0.92)	0.18	(Joy, 0.85)	0.20	0.15
2	(Humor, 0.93)	0.16	(Joy, 0.86)	0.22	0.17
3	(Neutral, 0.72)	0.45	(Neutral, 0.70)	0.55	0.48
4	(Serious, 0.89)	0.71	(Focused, 0.91)	0.78	0.80
5	(Serious, 0.95)	0.69	(Neutral, 0.88)	0.82	0.85

Table 4: Data extracted from the designed dialogues using the preference extractor in PAMU.

places dynamically updated preference with a fixed vector averaged over the first five turns, simulating static memory systems.

Experimental results are shown in Table 5, it can be seen that each module plays an essential and non-redundant role in maintaining consistency, personalization, and preference alignment throughout long-term interactions.

Methods	RA.†	MB.†	MG.†	AM.†
w/o. SW	11.24	12.03	10.07	15.36
w/o. EMA	11.35	12.47	10.78	14.05
Equal Fusion	13.56	16.45	15.43	20.34
w/o Detection	12.34	13.28	12.24	16.24
w/o Prompt	11.13	12.25	9.37	15.45
Single Pref	12.21	16.78	14.23	18.95
Static Pref	12.34	16.21	13.24	19.47
Full	**15.56**	**18.50**	**16.56**	**22.35**

Table 5: Ablation Study. We select the experimental results of temporal reasoning using LLaMA 7B and took the average of F1 and BLUE-1.

5 Further Analysis

To further demonstrate the interpretability and responsiveness of our Preference-Aware Memory Update (PAMU) mechanism, we design a controlled dialogue-based case study simulating a typical shift in user preference.

Since our mechanism is subjective and there is no objective metric to evaluate the specific effectiveness of its components, we incorporate both GPT-4 automatic scoring and human judgment. The human evaluation was conducted by ten annotators with bachelor's degrees, completed over a two-week period. Annotators were instructed not to use any AI tools during the assessment to ensure manual, unbiased evaluation. Notably, none of the paper's authors participated in the evaluation phase, ensuring fairness and neutrality.

We design the following dialogue to evaluate whether the components in PAMU mechanism are effective.

As shown above, the user initially demonstrates a clear preference for humorous and concise responses (Turns 1–2), but this preference abruptly shifts toward formal and information-dense content starting from Turn 3. PAMU captures this shift in real time by monitoring the divergence between the short-term (SW) and long-term (EMA) estimates for each preference dimension. At Turn 3, the preference change signal $C_t^{(d)}$ surpasses the predefined threshold δ in multiple dimensions (e.g., tone, length, density), triggering an immediate update to the fused preference vector $\hat{w}_t$ and rewriting of the prompt. Preference Vector Dynamics (Extracted) is shown in Table 4.

Example Dialogue

Turn 1 (User): Hey, tell me something and funny!
Turn 2 (User): That's good! I like it short and fun.
Turn 3 (User): Actually, I have a serious task now. Can you be more detailed?
Turn 4 (User): I need a thorough explanation on quantum computing basics.
Assistant (with PAMU): Certainly. Quantum computing is based on quantum bits, or qubits...
Assistant: Sure! Here's another fact to brighten you...
Turn 5 (User): Please just give me clear facts.

To evaluate the utility of PAMU's dynamic prompting, we compare model outputs with and without PAMU at Turn 4. Without PAMU, the model continues generating light, humorous content, misaligned with the user's updated intent. In contrast, the PAMU-augmented response accu-

rately adapts in tone, density, and formality, reflecting a meaningful understanding of user behavior change, as shown in dialogue content. Additionally, results in Tables 6 and 7 further demonstrate the effectiveness of PAMU.

Turn	Align(1-5)	Cons.	Response
w/o PAMU	2.1/2.2	✗/✗	2-turn delay
with PAMU	4.8/4.5	✓/✓	Real-time

Table 6: Comparison results, scoring results are in the format of (GPT/Human). Cons. represents consistency.

Methods	w/o PAMU	with PAMU
SC. (%)	37/35	92/94
PD. (%)	48/45	97/95

Table 7: Comparison results, scoring results are in the format of (GPT/Human). SC.: Style Consistency, PD.: Preference detection.

This case study confirms that PAMU can dynamically track evolving user preferences, detect both abrupt and gradual changes, and trigger appropriate generation adaptations, leading to more personalized, user-aligned interactions.

6 Conclusion

We propose a Preference-Aware Memory Update Mechanism to address the limitations of existing memory systems in tracking evolving user preferences. By combining sliding window and exponential moving averages, our method captures both short-term dynamics and long-term trends. A formalized change detection signal—based on their divergence—triggers memory updates, enabling interpretable and adaptive preference-aware behavior.

7 Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant NO. 62576292), the Zhejiang Province Leading Geese Plan (2025C02025), the Science and Technology Program of Huzhou (Grant NOs. 2023GZ42 and 2024GZ09), and in part by the Yangtze Delta Region Institute (Huzhou) Guidance Fund of University of Electronic Science and Technology of China (Grant NO. U03210054).

8 Limitations

Dependency on External Classifiers for Preference Extraction The performance of the Preference-Aware Memory Update (PAMU) mechanism is contingent upon the quality of its underlying preference extractors. Our mechanism relies on a suite of pre-trained models to quantify dimensions such as emotional tone, information density, and formality. The overall system's accuracy is thus bottlenecked by the performance of these individual components. Errors or domain-mismatches in any of these classifiers will propagate directly into the preference vector, potentially leading to the generation of misaligned prompts and subsequent responses. Future iterations could explore end-to-end training of the preference extractor or the use of a unified model to capture these dimensions more holistically.

Brittleness of Prompt-Based Control and Loss of Granularity The framework's reliance on explicit prompt injection has its inherent constraints. While this approach offers model-agnosticism and interpretability, its effectiveness can be brittle and highly dependent on the instruction-following capabilities of the base LLM. Some models may over- or under-index on specific parts of the structured prompt, leading to inconsistent or exaggerated stylistic shifts. Moreover, the conversion of continuous preference scores into discrete textual labels (e.g., "brief," "moderate," "dense") inevitably leads to a loss of granularity. A promising direction for future work is to explore "soft" prompting methods or direct manipulation of model activations to achieve more fine-grained behavioral control.

Limited Scope of Modeled Preference Dimensions The current set of five preference dimensions, while representative, may not capture the full spectrum of user intent. Preferences can be more nuanced, encompassing aspects like reasoning style (e.g., step-by-step vs. direct), desired level of creativity, or persona. The current framework is not designed to model these more abstract attributes. Additionally, PAMU primarily assumes that preferences are observable through linguistic cues in the dialogue history, which might not hold for implicit or task-level preferences. Expanding the dimensionality of the preference vector and developing methods to infer more latent user goals are important steps for future research.

References

DeepSeek-AI. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, and 1 others. 2024. A survey on llm-as-a-judge. *arXiv preprint arXiv:2411.15594*.

Taicheng Guo, Xiuying Chen, Yaqi Wang, Ruidi Chang, Shichao Pei, Nitesh V Chawla, Olaf Wiest, and Xiangliang Zhang. 2024. Large language model based multi-agents: A survey of progress and challenges. *arXiv preprint arXiv:2402.01680*.

Xu Han, Tianyu Gao, Yuan Yao, Demin Ye, Zhiyuan Liu, and Maosong Sun. 2019. Opennre: An open and extensible toolkit for neural relation extraction. *arXiv preprint arXiv:1909.13078*.

Xu Huang, Weiwen Liu, Xiaolong Chen, Xingmei Wang, Hao Wang, Defu Lian, Yasheng Wang, Ruiming Tang, and Enhong Chen. 2024a. Understanding the planning of llm agents: A survey. *arXiv preprint arXiv:2402.02716*.

Xu Huang, Weiwen Liu, Xiaolong Chen, Xingmei Wang, Hao Wang, Defu Lian, Yasheng Wang, Ruiming Tang, and Enhong Chen. 2024b. Understanding the planning of llm agents: A survey. *arXiv preprint arXiv:2402.02716*.

Haolin Jin, Linghan Huang, Haipeng Cai, Jun Yan, Bo Li, and Huaming Chen. 2024. From llms to llm-based agents for software engineering: A survey of current, challenges and future. *arXiv preprint arXiv:2408.02479*.

Kuang-Huei Lee, Xinyun Chen, Hiroki Furuta, John Canny, and Ian Fischer. 2024. A human-inspired reading agent with gist memory of very long contexts. *Preprint*, arXiv:2402.09727.

Yuanchun Li, Hao Wen, Weijun Wang, Xiangyu Li, Yizhen Yuan, Guohong Liu, Jiacheng Liu, Wenxing Xu, Xiang Wang, Yi Sun, and 1 others. 2024. Personal llm agents: Insights and survey about the capability, efficiency and security. *arXiv preprint arXiv:2401.05459*.

Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. 2024. Evaluating very long-term conversational memory of llm agents. *arXiv preprint arXiv:2402.17753*.

Charles Packer, Vivian Fang, Shishir_G Patil, Kevin Lin, Sarah Wooders, and Joseph_E Gonzalez. 2023. Memgpt: Towards llms as operating systems.

Rana Salama, Jason Cai, Michelle Yuan, Anna Currey, Monica Sunkara, Yi Zhang, and Yassine Benajiba. 2025. Meminsight: Autonomous memory augmentation for llm agents. *arXiv preprint arXiv:2503.21760*.

Haoran Sun, Haoyu Bian, Shaoning Zeng, Yunbo Rao, Xu Xu, Lin Mei, and Jianping Gou. 2025. Datasetagent: A novel multi-agent system for auto-constructing datasets from real-world images. *Preprint*, arXiv:2507.08648.

Haoran Sun and Shaoning Zeng. 2025. Introspection of thought helps ai agents. *Preprint*, arXiv:2507.08664.

Haoran Sun, Shaoning Zeng, and Bob Zhang. 2026. H-MEM: Hierarchical memory for high-efficiency long-term reasoning in LLM agents. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 341–350, Rabat, Morocco. Association for Computational Linguistics.

Hao Tian, Can Gao, Xinyan Xiao, Hao Liu, Bolei He, Hua Wu, Haifeng Wang, and Feng Wu. 2020. Skep: Sentiment knowledge enhanced pre-training for sentiment analysis. *Preprint*, arXiv:2005.05635.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *Preprint*, arXiv:2302.13971.

Bing Wang, Xinnian Liang, Jian Yang, Hui Huang, Shuangzhi Wu, Peihao Wu, Lu Lu, Zejun Ma, and Zhoujun Li. 2023. Enhancing large language model with self-controlled memory framework. *arXiv preprint arXiv:2304.13343*.

Junchao Wu, Shu Yang, Runzhe Zhan, Yulin Yuan, Lidia Sam Chao, and Derek Fai Wong. 2025. A survey on llm-generated text detection: Necessity, methods, and future directions. *Computational Linguistics*, pages 1–66.

Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. 2025. A-mem: Agentic memory for llm agents. *arXiv preprint arXiv:2502.12110*.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, and 40 others. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Yifan Yao, Jinhao Duan, Kaidi Xu, Yuanfang Cai, Zhibo Sun, and Yue Zhang. 2024. A survey on large language model (llm) security and privacy: The good, the bad, and the ugly. *High-Confidence Computing*, page 100211.

Zeyu Zhang, Xiaohe Bo, Chen Ma, Rui Li, Xu Chen, Quanyu Dai, Jieming Zhu, Zhenhua Dong, and Ji-Rong Wen. 2024a. A survey on the memory mechanism of large language model based agents. *arXiv preprint arXiv:2404.13501*.

Zeyu Zhang, Xiaohe Bo, Chen Ma, Rui Li, Xu Chen, Quanyu Dai, Jieming Zhu, Zhenhua Dong, and Ji-Rong Wen. 2024b. A survey on the memory mechanism of large language model based agents. *arXiv preprint arXiv:2404.13501*.

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