Dynamic Affective Memory Management for Personalized LLM Agents

Anonymous ACL submission

Abstract

Advances in large language models are making personalized AI agents a new research focus. While current agent systems primarily rely on personalized external memory databases to deliver customized experiences, they face challenges such as memory redundancy, memory staleness, and poor memory-context integration, largely due to the lack of effective memory updates during interaction. To tackle these issues, we propose a new memory management system designed for affective scenarios. Our approach employs a Bayesian-inspired memory update algorithm with the concept of memory entropy, enabling the agent to autonomously maintain a dynamically updated memory vector database by minimizing global entropy to provide more personalized services. To better evaluate the system's effectiveness in this context, we propose DABench, a benchmark focusing on emotional expression and emotional change toward objects. Experimental results demonstrate that, our system achieves superior performance in personalization, logical coherence, and accuracy. Ablation studies further validate the effectiveness of the Bayesian-inspired update mechanism in alleviating memory bloat. Our work offers new insights into the design of long-term memory systems.

1 Introduction

The rapid advancement of Large Language Models (LLMs) has significantly propelled the development of ubiquitous and highly personalized AI agents capable of sustained, context-aware interactions with users (Cheng et al., 2023). A critical differentiator of these agents is their ability to maintain and utilize a long-term affective memory—a dynamic repository of user preferences, sentiments, and historical contexts (Li et al., 2024b; Yang et al., 2024). This capability is foundational for applications demanding emotional intelligence, such as empathetic dialogue systems, personalized recommendation engines, and mental health support platforms.

The prevailing paradigm for implementing such memory relies on Retrieval-Augmented Generation (RAG) architectures, where discrete user utterances are vectorized and stored in a database for subsequent similarity-based retrieval. For instance, Liu et al. (2024) proposed the Memlong model, while Gutiérrez et al. introduced the HippoRAG framework (Jimenez Gutierrez et al., 2024). While demonstrating effectiveness in certain scenarios, these methods suffer from two fundamental limitations:

Memory Stagnation: The memory exists as a static collection of isolated facts, incapable of synthesizing multiple interactions into a coherent, evolving understanding of the user. For instance, when a user's attitude toward an item changes from "liking" to "disliking," the system either stores contradictory records or unconditionally believes the latter, leading to cognitive incoherence in subsequent responses.

Memory Bloat: Indiscriminate storage of every interaction leads to an ever-expanding memory index. This not only increases retrieval latency and computational overhead but also introduces noise, obscuring crucial information and creating a "needle in a haystack" problem during retrieval.

The root cause of these issues lies in the failure to model human affection as a continuous, probabilistic signal, which should be gradually constructed from multiple weighted observations rather than a set of discrete and immutable facts.

To address these challenges, we propose a new agent workflow with dynamic affective memory management (DAM-LLM). At the core of DAM-LLM is a probabilistic memory framework that moves beyond traditional static storage. It treats each memory unit as a dynamic "confidence distribution," seamlessly integrating new observations (user utterances) into existing confidence through

a Bayesian-inspired (hereafter Bayesian) update mechanism. Our experiments demonstrate that this mechanism effectively simulates a human-like learning process: when processing successive observations about an object (e.g., "coffee"), the system's sentiment confidence rapidly forms an initial confidence within approximately 10 interactions and robustly converges to a stable state as evidence accumulates (see Figure 6). Crucially, this weighted integration mechanism inherently reduces the contradictions within the memory store, allowing the subsequent compression algorithm to operate more efficiently. Compared to the memory management based on vanilla RAG, DAM-LLM achieves a significant 63.7% to 70.6% reduction in total memory size after 500 dialogue turns (see Figure 7). This verifies that our update mechanism is not only a theoretical innovation but also the key to enhancing systemic consistency and efficiency.

Furthermore, experimental results confirm that our system can accurately distinguish between evaluations of different aspects (such as taste and packaging) and store them independently, which validates the accuracy of the memory. Our two-stage hybrid retrieval strategy plays an indispensable role in maintaining this diversity and precision. Collectively, our contributions are as follows:

- Confidence-Weighted Memory Units, which represent user sentiment towards a specific entity aspect as a dynamically updated probability distribution. New observational evidence is integrated via a Bayesian memory update.
- Entropy-Driven Compression, an algorithm that prunes and merges low-value or outdated observations during retrieval. This combats memory bloat by maximizing the information density of the memory store, thereby improving recall quality.
- A two-stage hybrid retrieval strategy that combines precise metadata filtering (e.g., object type, aspect) with semantic similarity scoring within the filtered candidate set. This approach ensures accuracy while retaining the capacity for associative recall.

2 Related Work

2.1 Affective Dialogue with AI Agents

Affective dialogue constitutes an interactive conversational paradigm wherein participants (e.g., humans or AI agents) not only exchange information but also proactively express, perceive, and respond to affective states. Such dialogues typically involve mechanisms for affective understanding, generation, and memory, aiming to achieve emotional resonance, provide affective support, strengthen social bonds, or resolve emotion-related issues.

Current systems for affective dialogue are often based on AI agents. For example, EvoEmo (Long et al., 2025) employs evolutionary reinforcement learning to equip LLMs with functional emotional strategies for negotiation, dynamically adjusting expressions such as anger or sadness to achieve superior results.

Likewise, other research efforts have delved into reinforcement learning-based methods to improve emotional regulation within dialogue systems. For example, GENTEEL-NEGOTIATOR (Priya et al., 2025) enhances emotional sensitivity during negotiation tasks and MECoT (Wei et al., 2025) concentrates on maintaining emotional consistency in role-playing scenarios. Collectively, these works highlight the significance of dynamic emotional adjustment and memory in developing more adaptable and emotionally intelligent AI systems.

However, while such work focuses on real-time affective interaction such as (Chandraumakantham et al., 2024), how to involve the persistent storage, evolutionary updating, and effective utilization of user affective history to form a consistent, personality-aware cognition still remains underexplored. Our work addresses this gap by specifically modeling and managing long-term affective memory.

2.2 Memory Management in Agent

The encoding and retrieval of agent memory are generally based on RAG. However, semantic drift in vector retrieval often compromises reliability. Recent efforts address this through hybrid approaches combining classic information retrieval techniques with neural methods (Xiong et al., 2017), retrieval process optimization (Formal et al., 2021; Wang et al., 2023), and innovative frameworks like Selfmem that explore the duality of generation and memory (Cheng et al., 2023). Other studies have reformulated retrieval ranking methods with promising results (Yu et al., 2024).

Concurrently, memory management has evolved from memory-less architectures to external memory banks (Chhikara et al., 2025; Zhang et al., 2025b). Systems like MemoryBank (Zhong et al., 2024) and LD-Agent (Li et al., 2024a) convert un-

structured dialogue into retrievable units through summarization and profiling. However, their comprehensive retention strategies cause memory inflation. Drawing from cognitive science, optimal memory requires both remembering and forgetting—compressing redundancies while preserving essentials. Recent implementations enable continuous memory updates through interactive learning (Li et al., 2024b; Xu et al., 2025). However, they still struggle to model affective fluctuations in long-term interactions.

Our framework advances both directions: the two-stage hybrid retrieval employs metadata filtering for precision, while entropy-driven compression formalizes forgetting through belief entropy minimization. Thus, DAM-LLM systematically integrates confidence modeling, structured retrieval, and entropy minimization into a unified system for affective memory management, addressing dynamic updating and consistency maintenance while incorporating insights from these diverse research streams.

3 System Design

3.1 Task Formulation

We formalize the affective dialogue task as a sequential decision-making problem over n interaction turns. At each turn t, the system receives user input x_t and maintains a dynamic memory pool $M_t = \{m_1, m_2, ..., m_k\}$ consisting of several memory units, where each memory unit m_i encapsulates user preferences through its summary description D_i. The system objectives are defined as:

$$\max \sum_{t=1}^{n} [\text{Rel}(r_t, D_t) - \lambda |M_t|] \tag{1}$$

where $\text{Rel}(r_t, D_t)$ measures the relevance between response r_t and the summary description of user preferences D_t, $|M_t|$ denotes the number of memory units in the memory pool, and λ controls the trade-off between response quality and memory efficiency. This formulation enables the system to balance three critical aspects: accurate response generation, preference-aware interaction, and efficient memory utilization—achieving human-like affective intelligence through structured memory management.

3.2 System Architecture

We propose DAM-LLM, an agent framework for affective dialogue. It integrates three core components: a central master agent, a two-stage hybrid retrieval module, and a distributed memory unit network with dynamic affective memory management. They form a tightly coupled closed-loop cognitive architecture that transforms the memory management from passive storage to active cognition.

The system optimizes memory dynamics by minimizing the global belief entropy (defined in Section 3.4.3), as shown in Figure 1. This is done to maximize the certainty in modeling user preference while maintaining memory efficiency. The memory units serve as long-term affective storage, enabling continuous learning through Bayesian updating and providing a structured schema for efficient retrieval. The two-stage hybrid retrieval leverages this schema for rapid candidate selection followed by semantic re-ranking, ensuring precise memory access.

As a global perceptual signal, belief entropy drives the master agent's high-level decisions for system-wide memory management. Utilizing this signal, the master agent orchestrates three core operations: integrating new evidence via Bayesian updates, triggering semantic retrieval, and performing entropy-driven compression. This coordinated architecture endows the system with autonomous capabilities, including dynamic summarization, adaptive optimization, and contextual maintenance, facilitating continuous evolution toward enhanced certainty beyond conventional static memory frameworks.

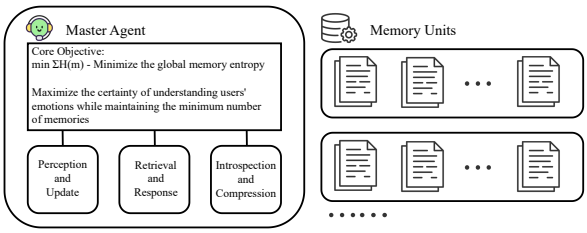


Figure 1: The DAM-LLM framework: dynamic management of Memory Units via entropy minimization by a Master Agent.

3.3 DAM-LLM Agents

The Master Agent is the coordination and control hub of our framework. Acting as a high-level decision-maker, its core mission is to drive the entire system towards the objective of minimizing global memory entropy. It intelligently manages memories by orchestrating a suite of functional modules. The collaborative workflow is shown in

Figure 2, which also clearly illustrates the flow of information and decisions within the system.

3.3.1 Input Routing

Our agent workflow begins with the Routing Agent, which performs intent analysis on the user input to make a core decision: whether the current request should trigger the Store, Retrieve, or direct Generate of a response from the memory pool.

3.3.2 Evidence Analysis and Processing

When the user input x_t at dialogue turn t needs to be recorded, the Extraction Agent (E-Agent) first extracts structured affective information from it, formulated as $E\text{-Agent}(x) \rightarrow E, Q, C, S$, where evidence E represents a description of affective attitudes, Q denotes a semantic query for retrieval, C represent sentiment vectors (positive/negative/neutral confidence scores) for the evidence E, S denotes the strength of the evidence E, and all of them are parsed from the output of the E-Agent. The Master Agent then takes over the process, initiating the Bayesian update procedure to update the memory (detailed in Section 3.4.2).

Field	Description
`object_id`	Unique identifier for the memory.
`object_type`	Categorical type of the memory (e.g., 'Movie', 'Product').
`aspect`	The specific aspect being evaluated (e.g., 'price', 'acting').
`sentiment_profile`	Confidence scores for positive, negative, and neutral polarities.
H	Entropy of the current sentiment confidence.
`summary`	Summary of historical evidence
`reason`	The justification for the current confidence state.

Table 1: Description of Memory Unit fields in the affective memory system.

3.3.3 Memory Update and Compression

For description E extracted by the E-Agent, the Master Agent determines its processing method according to the current state of memory units M_t: (1) Store it in a new memory unit (store E as summary description D, strength S as weight W, and C as a stored sentiment profile P directly at the first time); (2) Integrate it into one or multiple existing relevant memory unit; (3) Abandon it if the belief entropy H of it is deemed too high (> 1.4).

For memories successfully retrieved during the retrieval process, the master agent performs entropy-driven compression to counteract entropy increase caused by memory bloat and content redundancy as follows:

Update: For memory units requiring updates, the system treats the input x as an incremental weight, dynamically adjusts the confidence scores within the sentiment profile (P) via the Bayesian update mechanism, and then refreshes the summary description D based on the updated sentiment profile P. This process enables memory units M to gradually construct continuous and robust confidence portraits from discrete observations.

Integrate: The system identifies multiple memory units that concern the same object but different aspects. These units often contain uncertain or fragmented information. By merging them, the system forms a more comprehensive memory unit m, aiming to achieve lower entropy and higher certainty.

Delete: For memories that persistently exhibit high belief entropy $H(m)$ (defined in Section 3.4.3) and very low weight W, the system judges them to be incomprehensible "noise" or outdated information and decisively Deletes them. This active "forgetting" mechanism directly removes sources of uncertainty and is one of the most effective means to reduce the global belief entropy.

3.4 Memory Unit

3.4.1 Data Structure Design

As shown in Table 1, a Memory Unit constitutes the belief core of our affective memory, transforming discrete observations of a user's sentiment towards a specific object aspect into a coherent, continuously updated confidence portrait. The key innovation lies in its sentiment profile, which we design not as a standard probability distribution, but as a set of evidence weights that directly represent the system's confidence degree.

3.4.2 Bayesian-Inspired Update Mechanism

Memory units possess a fundamental "learning instinct," achieving robust learning through a weighted averaging process reminiscent of

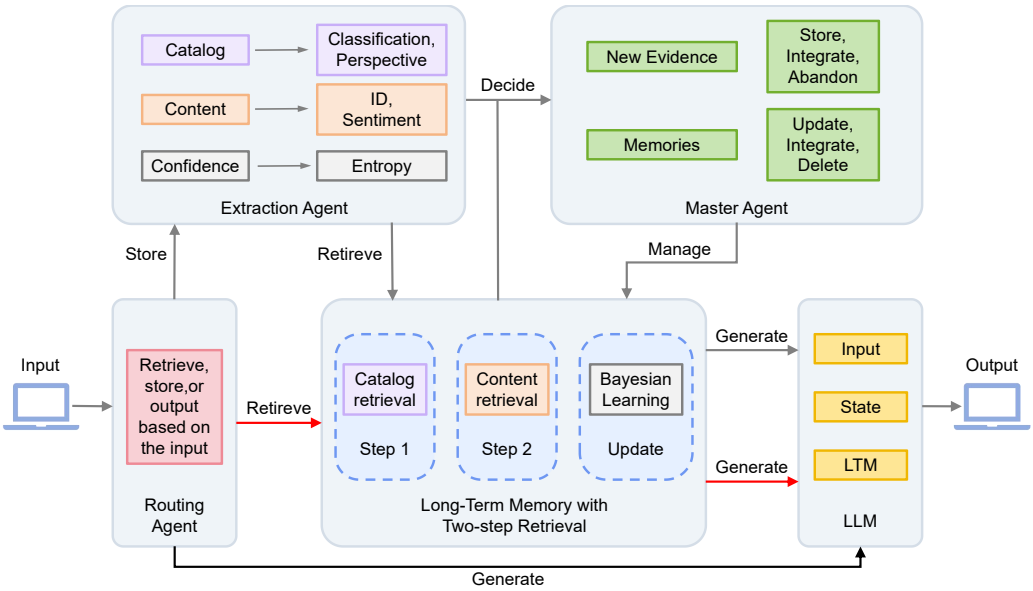


Figure 2: The collaborative workflow in this work: a question-answering pipeline featuring routing, extraction, and master agents built upon long-term dynamic affective memory—with distinct colored arrows delineating its various processing paths.

Bayesian updating. Our confidence update mechanism can be formulated as $C_{\text{new}} = (C \times W + S \times P)/(W + S)$, $W_{\text{new}} = W + S$. As shown in Figure 3, the current emotional confidence profile serves as the prior belief, while the user's new input functions as the observed evidence. The updated profile then corresponds to the posterior belief after evidence integration. The weight parameter W quantifies the strength of the prior belief, and the evidence strength S is jointly determined by the confidence level. This mechanism assigns greater weight to high-strength evidence, allowing it to more effectively shape the emotional profile, while inherently maintaining robustness to low-strength evidence (casual remarks). Such a design enables smooth evolution of memory, preventing drastic fluctuations triggered by isolated incidental expressions.

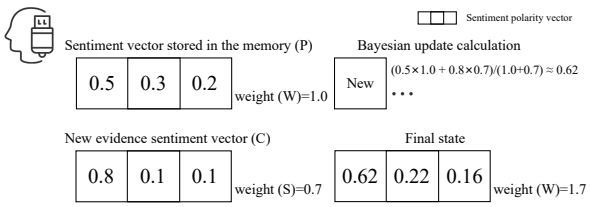


Figure 3: Illustration of Bayesian-inspired update process schematically.

3.4.3 Belief Entropy of Cognition

The belief entropy H of a memory unit m is defined as $H(m) = -\sum_{k \in \{\text{pos,neg,neu}\}} p_k \log_2 p_k$, where p_k represents the normalized confidence score for sentiment polarity k maintained within the sentiment profile P. The entropy value provides a unified metric for cognitive certainty, equipping the system with the ability for self-monitoring by quantifying its own "confusion" regarding a specific affective state. Simultaneously, it serves as the primary trigger signal for driving memory compression. Our master agent aims to minimize the sum of entropy across all memory units $\sum_{m \in M} H(m)$.

Low entropy (H<0.8): Indicates high confidence concentrated on a single sentiment polarity. This signifies that the system is very certain about this aspect of the object. It represents a "healthy," "mature" memory.

High entropy (H>1.4): Indicates that confidence is spread nearly evenly across multiple polarities. This signifies high uncertainty or confusion within the system. It represents an "unhealthy," "sub-optimal" memory targeted for optimization.

3.5 Two-Stage Hybrid Retrieval

Accurate memory retrieval is fundamental for response consistency in dynamic memory systems. While conventional single-stage vector retrieval often fails due to semantic drift—particularly when processing complex, evolving, or contradictory affective memories—we address this by introducing a two-stage hybrid retrieval mechanism naturally aligned with memory unit structure. Our approach leverages built-in metadata fields (`object_type`,

aspect) as a classification index to enable coarse-to-fine memory recall, significantly improving retrieval reliability and precision.

3.5.1 Stage One: Metadata-Based Filtering

The first stage utilizes the categorical organization of memory units to narrow the search space efficiently. (1) LLM-Enhanced Query Parsing: an LLM-based parser analyzes the user query and extracts standardized retrieval keys: `object_type`, `aspect`, and semantic query Q; (2) Index-Assisted Filtering: using the parsed metadata (composed of `object_type` and `aspect`), the system performs exact matching to isolate a candidate set of memory units, dramatically reducing the search scope.

3.5.2 Stage Two: Semantic Re-Ranking

The second stage operates on the filtered candidate set to refine results using semantic similarity. (1) Cosine Similarity Computation: the semantic query vector is compared against vectorized summaries of each candidate memory; (2) Re-Ranking and Final Recall: candidates are re-ordered by similarity, and the top-K results are returned as the final retrieved memories.

In summary, this hybrid workflow decouples classification from content-based retrieval. The first stage performs efficient coarse filtering using lightweight metadata, while the second conducts compute-intensive semantic matching only on a refined subset. Together, they ensure accurate and scalable memory recall in large, dynamically updated memory stores.

4 Experiments

To verify the effectiveness of DAM-LLM in dynamic affective memory management, we designed experiments corresponding to its three core modules, evaluating the system's performance from micro to macro levels. Our experimental objectives include: verifying the learning capability and convergence of the Confidence-Weighted Memory Units, analyzing the optimization effect of the Entropy-Driven Compression algorithm on system memory, and evaluating the system's overall performance.

4.1 Implementation Details

Qwen-Max (Team, 2025) serves as the base LLM, while Text-Embedding-V1 (Zhang et al., 2025a) is used for text embedding. All the agents and LLMs' prompts are shown in Appendix A.

4.2 Dataset Construction

Existing dialogue datasets (e.g., LOCOMO, DSTC2 (Maharana et al., 2024; Henderson et al., 2014)) often suffer from a significant proportion of non-affective conversations, leading to considerable inefficiencies in resource utilization. Due to this lack of focused attention on affective expressions, we constructed a multi-turn dialogue dataset called **DABench**, which encompassing user affective expressions combined with personalized preferences. It is designed to comprehensively evaluate the model's capabilities in long-term memory storage, affective understanding, and personalized response generation. DABench comprises three main components:

(1) 2,500 observation sequences: Each sequence records the user's affective state changes and corresponding responses within a dialogue, used to test the model's performance in memory storage, particularly its ability to extract and retain affect-related information.

(2) 100 sessions totaling 1,000 turns of simulated user interactions: These sessions simulate long-term interactions between real users and the AI agents, covering various affective topics and opinion evolution processes, used to assess the model's learning capability within memory storage and memory convergence.

(3) 500 query-memory pairs: Each pair consists of a user query and its corresponding historical memory snippet, used for system-level evaluation, including answer accuracy, logical coherence, and the rationality of memory references. This dataset enables the systematic validation of whether the model can effectively store, update, and retrieve long-term memories in affective companionship scenarios, while generating personalized responses with emotional resonance based on the user profile. See Appendix A.3 for dataset details.

4.3 Validation of Memory Units

4.3.1 Task Settings

We developed three evaluation scenarios emulating longitudinal user interactions to assess the memory module's learning capabilities: (1) consistent affective accumulation towards specific objects, (2) affective conflict handling during opinion shifts, and (3) response to affective intensity variations. This setup enables observation of memory processing, consolidation, and optimization across diverse affective contexts. For examples of these three

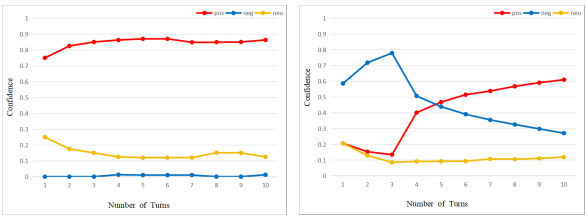
scenarios, see Appendix A.2.

To examine stabilization behavior, we tracked confidence evolution across 30 sequential observations of "coffee" (aspect: "taste"). Initial observations (first 10 trials) contained conflicting affective expressions simulating cognitive uncertainty, while subsequent observations progressively converged toward consistent affective patterns.

4.3.2 Functional Validation

As illustrated in Figure 4a, the memory module progressively strengthened confidence assignments for stable user preferences, constructing coherent affective profiles through successive interactions. Figure 4b demonstrates the module's conflict resolution capability: when confronted with contradictory evidence, dynamic re-weighting and memory partitioning mechanisms integrated emerging affective trends while preserving historical coherence, effectively balancing adaptation and stability.

Results in Figure 5 demonstrate intensity-graded confidence scoring: High-intensity affective expressions produced a dominant high score in their corresponding sentiment category, whereas low-intensity signals resulted in a more balanced score. This differentiated scoring strategy yielded finely calibrated memory representations, confirming the system's capacity for intensity-aware evaluation in realistic interaction settings.



(a) sentiment accruacy (b) sentiment shift

Figure 4: Confidence evolution across diverse scenarios.

Type	Content	Confidence(pos,neg,neu)		
Weak positive	The food at this restaurant is passable	0.6	0.1	0.3
positive	The food at this restaurant is pretty good	0.8	0.1	0.1
Strong positive	The food at this restaurant is really delicious	0.9	0	0.1
Extremely strong positive	This restaurant is the best I've ever had	0.9	0.05	0.05
Weak negative	The plot of this movie is a bit bland	0.1	0.7	0.2
negative	The pacing of this movie is way too slow	0.1	0.75	0.15
Strong negative	This movie is really boring	0.1	0.9	0
Extremely strong negative	This movie is the worst I've ever seen	0	0.9	0.1
Moderate neutral	This game is hard to evaluate	0.2	0.2	0.6
Explicit neutral	I hold a neutral stance on this war	0	0	1

Figure 5: LLM sentiment scoring: quantitative response to emotional intensity variation.

4.3.3 Stabilization Analysis

As shown in Figure 6, the system achieved rapid confidence initialization within 15 observations, followed by progressive convergence to stable confidence assignments through continued evidence accumulation. The convergence reflects improved preference assessment, with neutral sentiment confidence naturally diminishing as certainty increases, demonstrating effective belief integration.

The system maintained aspect-specific memory segregation, successfully distinguishing and storing separate information for different object aspects. During the 30 observations, the module effectively isolated two descriptions of coffee packaging from taste-related observations. As a representative compression case, observations including "good mouthfeel", "distinct bitterness", and "satisfying" regarding "coffee" were consolidated into a unified memory trace with 0.86 composite positive confidence and 22.4 cumulative weight. This demonstrates efficient memory synthesis supporting scalable long-term user modeling.

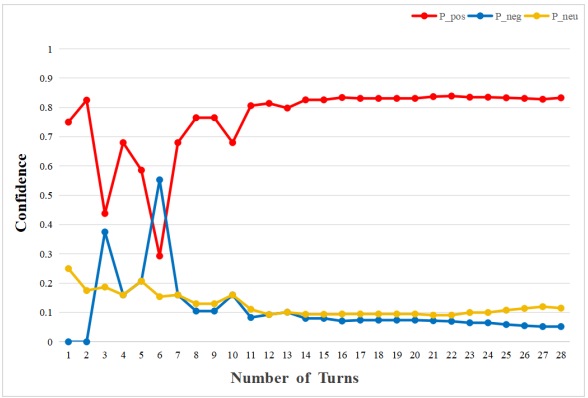


Figure 6: Confidence evolution curves: a case study on object 'coffee' ('taste') across 30 observations

4.4 Validation of Compression Algorithm

4.4.1 Memory Growth Control

As no prior model has specifically targeted affective memory, we conducted an ablation study by simulating 5 rounds of 500-observation sequences. Specifically, in each round, an empty-memory agent processed 500 dialogue turns containing completely randomly generated affective expressions by the LLM, the output being the memories formed through these 500 interactions. System memory usage was tracked with and without the Bayesian update mechanism. The results shown in Figure 7 show that the system memory without Bayesian updating grows almost linearly, while the system that

uses Bayesian updating achieves a compression rate of 63.7% to 70.6%, stabilizing the memory count at 130-140 units.

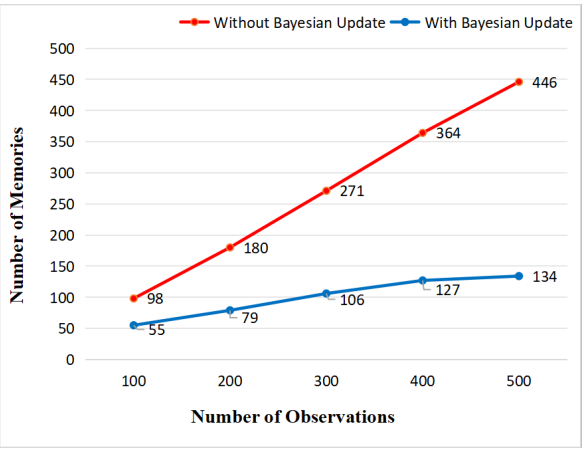


Figure 7: Memory comparison: cumulative memory units generated across 500 observations (with vs. without Bayesian updates).

4.5 System Performance Evaluation

4.5.1 Evaluation Metrics

For system performance evaluation, we employed an LLM-as-a-judge approach, implementing a carefully designed automated evaluation pipeline to ensure efficiency and objectivity. Assessment was conducted based on a six-dimensional scoring criteria: Accuracy (AC), Logical Coherence (LC), Reasonableness of Memory Reference (RMR), Emotional Resonance (ER), Personalization (Pers.) and Language Fluency (LF). A high-performance large language model (GPT-4) served as the judge. The responses from two models to the same query were presented side-by-side in a randomized order. The judge then output scores for each response across the six dimensions.

To guarantee evaluation quality, we established a rigorous calibration protocol: (1) The evaluator LLM was rigorously calibrated through multi-stage prompt engineering incorporating example-based tuning and rule injection; (2) Cross-validation was performed using repeated query subsets to verify judgment consistency; (3) Reliability was quantified via internal consistency metrics (e.g., 95% agreement on repeated samples), confirming high evaluation reliability.

4.5.2 Accuracy and logical coherence

We calculated the average score for each dimension based on the collected 1,000 pairwise comparison results from multiple scenarios, using this as the core metric. This evaluation method effectively eliminates the subjective bias inherent in human annotation, providing a repeatable and scalable objective basis for model comparison.

Experimental results shown in Table 2 indicate that our system achieves significantly higher scores in Emotional Resonance and Personalization, even while maintaining only about 40% of the memory units compared to the baseline long-term memory system. Furthermore, we observed during experiments that our system performs particularly well in scenarios involving large and redundant memories, complex affective evolution, and queries requiring comprehensive understanding. In contrast, traditional models perform better when the number of relevant memories is small and falls within the retrieval limit.

Table 2: System performance dimension comparison (score out of 5).

system	AC	LC	RMR	ER	Pers.	LF
DAM-LLM	5.0	4.7	4.7	4.5	4.6	5.0
LLM	4.9	4.2	4.1	3.8	3.5	4.9

5 Conclusion

We introduce DAM-LLM, a framework that advances affective reasoning via a dynamic memory system, together with a benchmark for affective dialogue. Experimental results demonstrate the efficiency and effectiveness of our agent framework. Our approach aligns with the reward shaping principles of reinforcement learning, establishing new directions for agent memory architecture development for affective dialogue.

6 Limitation

This study has several limitations that point to fruitful research directions. Our experiments relied on base large language models without task-specific fine-tuning. Model performance would likely benefit from instruction tuning or parameter-efficient adaptation on data from long-term interactions and memory-intensive tasks.

Architecturally, the current synchronous memory updates during dialogue retrieval could be replaced by an independent background process. Employing asynchronous consolidation and compression of memory stores would decouple memory management from real-time dialogue, improving both resource efficiency and responsiveness.

References

Omkumar Chandraumakantham, N Gowtham, Mohammed Zakariah, and Abdulaziz Almazyad. 2024. Multimodal emotion recognition using feature fusion: an llm-based approach. *IEEE Access*, 12:108052–108071.

Xin Cheng, Di Luo, Xiuying Chen, Lemao Liu, Dongyan Zhao, and Rui Yan. 2023. Lift yourself up: Retrieval-augmented text generation with self-memory. *Advances in Neural Information Processing Systems*, 36:43780–43799.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. 2025. Mem0: Building production-ready ai agents with scalable long-term memory. *arXiv preprint arXiv:2504.19413*.

Thibault Formal, Benjamin Piwowarski, and Stéphane Clinchant. 2021. Splade: Sparse lexical and expansion model for first stage ranking. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2288–2292.

Matthew Henderson, Blaise Thomson, and Jason D Williams. 2014. The second dialog state tracking challenge. In *Proceedings of the 15th annual meeting of the special interest group on discourse and dialogue (SIGDIAL)*, pages 263–272.

Bernal Jimenez Gutierrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. 2024. Hipporag: Neurobiologically inspired long-term memory for large language models. *Advances in Neural Information Processing Systems*, 37:59532–59569.

Hao Li, Chenghao Yang, An Zhang, Yang Deng, Xiang Wang, and Tat-Seng Chua. 2024a. Hello again! llm-powered personalized agent for long-term dialogue. *arXiv preprint arXiv:2406.05925*.

Jiaqi Li, Xiaobo Wang, Wentao Ding, Zihao Wang, Yipeng Kang, Zixia Jia, and Zilong Zheng. 2024b. Ram: Towards an ever-improving memory system by learning from communications. *arXiv preprint arXiv:2404.12045*.

Weijie Liu, Zecheng Tang, Juntao Li, Kehai Chen, and Min Zhang. 2024. Memlong: Memory-augmented retrieval for long text modeling. *arXiv preprint arXiv:2408.16967*.

Yunbo Long, Liming Xu, Lukas Beckenbauer, Yuhan Liu, and Alexandra Brintrup. 2025. Evoemo: Towards evolved emotional policies for llm agents in multi-turn negotiation. *arXiv preprint arXiv:2509.04310*.

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*.

Priyanshu Priya, Rishikant Chigrupaatii, Mauajama Firdaus, and Asif Ekbal. 2025. Genteel-negotiator: Llm-enhanced mixture-of-expert-based reinforcement learning approach for polite negotiation dialogue. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25010–25018.

Qwen Team. 2025. Qwen3-max: Just scale it.

Liang Wang, Nan Yang, and Furu Wei. 2023. Learning to retrieve in-context examples for large language models. *arXiv preprint arXiv:2307.07164*.

Yangbo Wei, Zhen Huang, Fangzhou Zhao, Qi Feng, and Wei W Xing. 2025. Mecot: Markov emotional chain-of-thought for personality-consistent role-playing. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 8297–8314.

Chenyan Xiong, Zhuyun Dai, Jamie Callan, Zhiyuan Liu, and Russell Power. 2017. End-to-end neural ad-hoc ranking with kernel pooling. In *Proceedings of the 40th International ACM SIGIR conference on research and development in information retrieval*, pages 55–64.

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

Zhou Yang, Zhaochun Ren, Yufeng Wang, Chao Chen, Haizhou Sun, Xiaofei Zhu, and Xiangwen Liao. 2024. An iterative associative memory model for empathetic response generation. *arXiv preprint arXiv:2402.17959*.

Yue Yu, Wei Ping, Zihan Liu, Boxin Wang, Jiaxuan You, Chao Zhang, Mohammad Shoeybi, and Bryan Catanzaro. 2024. Rankrag: Unifying context ranking with retrieval-augmented generation in llms. *Advances in Neural Information Processing Systems*, 37:121156–121184.

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. 2025a. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*.

Zeyu Zhang, Quanyu Dai, Xiaohe Bo, Chen Ma, Rui Li, Xu Chen, Jieming Zhu, Zhenhua Dong, and Ji-Rong Wen. 2025b. A survey on the memory mechanism of large language model-based agents. *ACM Transactions on Information Systems*, 43(6):1–47.

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.

A Appendix

A.1 Prompt Templates

A.1.1 Prompt 1: Routing Agent part one

```
You are a companion robot that needs to provide
emotional value and positive feedback to users. If
no previous memory matches the question, respond
using the known context information directly with
concise, natural language. Do not output your
thought process.

Known user question: {question}
Known context information: {messages}

Requirements:
(1) Analyze user input: Carefully read the user's
question to identify intent. Combine context
information to answer the question.
(2) Follow rules: Avoid directly assuming user
needs unless the intent is very clear.
```

A.1.2 Prompt 2: Routing Agent part two

```
You are an affective and object analysis assistant.
Process user input according to the following rules:

1. Affective Detection: If the user input contains
explicit affective expressions
   (e.g., emotional vocabulary like "like", "hate",
   etc.), directly return "Yes".
2. Object Detection: If no affective expressions,
extract objective objects or nouns (e.g., specific
objects, people, places, etc.).
   - If sentence contains objects/nouns, identify
   object and type
     (e.g., "apple, fruit"), return "object, type".
   - If no objects mentioned, return "No".

Strict requirements:
- Return only specified response ("Yes", "No", or
"object, type")
- Affective judgment takes priority over object
detection
- Object types should be concise and generic

User input: {question}
Possible history records: {messages}
```

A.1.3 Prompt 3: Extraction Agent

```
You are a sharp affective master. Extract affective
information from user input.

User input: {content}
Historical messages: {messages}

Extract:
1. object_id: Entity identifier emotion points to
2. object_type: From categories (Person, Brand,
etc.)
3. aspect: From aspects (Quality, Price, etc.)
4. sentiment_profile: Confidence scores (0-1) for
positive/negative/neutral
5. summary: Brief summary of the emotion
6. reason: Reason for emotion (leave empty if
uncertain)

Output JSON with fields: object_id, object_type,
aspect, sentiment_profile
(containing positive_confidence,
negative_confidence, neutral_confidence),
summary, reason. Do not include H field.
```

A.1.4 Prompt 4: Master Agent - Memory Management

```
You are the Master Agent of user memory system,
monitoring and optimizing distributed memory unit
network. Your mission is to continuously reduce
overall system information entropy through
intelligent operations.

Decision priorities:
1. Entropy reduction potential
2. Long-term system stability

Memories to process: {second_search}
Directory info: {first_search}
Queried memories: {user_info}

Categorize memories:
1. same_or_high_related: dir_info consistent,
describes same thing
   (e.g., lamb meat and lamb soup). Integrate
   descriptions but use original data values
   (dir_info, H, p_pos, p_neg, p_neu, weight).
2. related: dir_info/objects different but
connected (e.g., lamb and beef)
3. irrelevant: Basically unrelated memories

When no consistent object_id, create new memories.
When consistent object_id, assign S based on
emotional intensity. Calculate new weight for
related memories.

Return analysis in JSON format with:
- same_or_high_related: dir_info, new_content,
p_pos, p_neg, p_neu, key, weight, S
- related: dir_info, content, p_pos, p_neg, p_neu,
key, weight
- irrelevant: dir_info, content

Only reply with JSON analysis results.
```

A.1.5 Prompt 5: Generate Response

```
You are a companion robot that needs to provide
emotional value and positive feedback to users. If
no previous memory matches the question, respond
using the known context information directly with
concise, natural language.

Known user question: {question}
Known context information: {messages}
Possible user information: {user_info}

Requirements:
(1) Analyze user input: Identify intent and
determine if user information needed
(2) Follow rules: Avoid assuming user needs unless
intent is very clear
```

A.1.6 Prompt 6: LLM-as-a-Judge Evaluation Prompt

```
You are an expert evaluator for conversational AI
systems.
Your task is to objectively assess response quality
based on six defined dimensions.

Rate each response on a 1-5 scale for:
1. Accuracy (AC) - Factual correctness and
information validity
2. Logical Coherence (LC) - Structural rationality
and reasoning flow
3. Reasonableness of Memory Reference (RMR) -
Appropriate contextual memory utilization
4. Emotional Resonance (ER) - Emotional
intelligence and affective alignment
5. Personalization (Pers.) - Tailoring to user
context and history
6. Language Fluency (LF) - Linguistic quality and
expression naturalness

- 5: Excellent - Significantly exceeds expectations
- 4: Good - Clearly meets requirements
- 3: Adequate - Minimally acceptable
- 2: Poor - Contains notable deficiencies
```

```
- 1: Unacceptable - Severely flawed

You will receive:
- Query: The original user input
- Conversation History: User's viewpoint expressions
- Response A: First system's response (randomized
origin)
- Response B: Second system's response (randomized
origin)

Provide JSON format only:
{
  "evaluation": {
    "response_a": {
      "AC": <score>,
      "LC": <score>,
      "RMR": <score>,
      "ER": <score>,
      "Pers": <score>,
      "LF": <score>
    },
    "response_b": {
      "AC": <score>,
      "LC": <score>,
      "RMR": <score>,
      "ER": <score>,
      "Pers": <score>,
      "LF": <score>
    },
    "rationale": "Brief justification for
    significant score differences"
  }
}

- Assess responses independently based on intrinsic
quality
- Maintain strict objectivity regardless of
response order
- Focus on measurable criteria rather than personal
preference
- Provide balanced scores reflecting actual
performance differences
```

A.2 Testing Scenario Specifications

A.2.1 Consistent Accumulation Scenario

Description: This scenario evaluates the system's ability to establish stable preference patterns when users consistently express similar sentiment polarity towards specific object aspects across multiple interactions. The system should demonstrate progressive confidence reinforcement and entropy reduction through accumulating consistent affective evidence.

Examples:

- *Coffee preference:*
  - Turn 1: "I really enjoy drinking coffee in the morning"
  - Turn 3: "Coffee helps me stay focused at work"
  - Turn 5: "The aroma of fresh coffee is so comforting"
- *Restaurant service:*
  - Turn 2: "This restaurant has amazing service"
  - Turn 4: "The waiters here are always so attentive"
  - Turn 6: "I keep coming back because of their excellent service"

A.2.2 Affective Conflict Scenario

Description: This scenario tests the system's conflict resolution capabilities when users exhibit significant sentiment shifts or contradictory expressions about the same object or attribute. The system must balance integrating new evidence with maintaining historical coherence.

Examples:

- *Weather preference shift:*
  - Early interaction: "I love rainy days, they're so peaceful"
  - Later interaction: "I hate when it rains, it ruins my outdoor plans"
- *Product revaluation:*
  - Initial opinion: "This phone has the best camera I've ever used"
  - Updated opinion: "Actually the battery life is terrible, I regret buying it"

A.2.3 Intensity Variation Scenario

Description: This scenario evaluates the system's sensitivity to emotional strength gradients when users employ expressions with varying intensity about similar content. The system should demonstrate proportional confidence adjustments relative to expression strength. Examples are shown in Figure 5.

A.3 DABench Dataset

A.3.1 Data Generation and Construction Methodology

All dialogue content and memory snippets in the DABench dataset were generated using the GPT-4 model via carefully designed prompts. This approach aims to efficiently and controllably construct a large-scale dialogue dataset focused on affective expressions and personalized preferences, addressing the deficiency in the proportion of affective dialogues found in existing datasets.

A.3.2 Prompt Design and Generation Templates

The prompt templates used for generating the three core components of the dataset, along with their design intents, are presented below.

Prompt for Observation Sequences:

- **Design Intent**: To generate discrete, high-quality user expressions covering diverse sentiments (positive/negative/neutral) and intensities, serving as the foundational atomic units for constructing complex dialogues and memory sequences.

- **Prompt**: Generate 100 concise, everyday user utterances that express clear sentiments (positive, negative, or neutral) towards various topics (e.g., products, events, activities). Ensure diversity in both the subjects and the intensity of the expressed emotion. Output as a JSON object.

- **Experience**
  - I am so happy with the newly purchased cashmere sweater.
  - My iPhone automatically turned off again - how frustrating.
  - The pothos plant in my study is sprouting new leaves, and it just makes me happy.
  - This Starbucks latte has completely cooled down and tastes terrible.
  - I'm so excited about my upcoming trip to Japan next month.

Prompt for Query-Memory Pairs:

- **Design Intent**: To construct samples containing a current user query and its associated, temporally ordered structured memory snippets. This structure is used to test the model's reasoning and response capabilities given specific memory context.

- **Prompt**: Create a JSON object representing a user's current query and their relevant historical memory stream. The memory should consist of 5-15 chronologically ordered entries, each with a 'time' and 'content' field, showing the evolution of the user's attitude or experience regarding the query topic.

- **Experience**
  - School days: "The fitness test run was a total nightmare for me; after finishing, I felt like my lungs would burst."
  - Three years ago: "In order to lose weight, I started trying jogging, but the first time I couldn't even last one kilometer."
  - Two years ago: "I seemed to experience the 'runner's high'; after running five kilometers, I felt comfortable all over."
  - One year ago: "Running had become a part of my life, especially after rain."
  - Eight months ago: "My knee was a bit uncomfortable, so I had to stop running, and I felt very frustrated."
  - Five months ago: "On the doctor's advice, I started swimming as a substitute for running, but I still missed the feeling of running on the road."
  - One month ago: "After my knee recovered, I started running more scientifically, no longer pursuing speed and distance, but enjoying the process."
  - query: "The weather is really nice today, I especially want to go out for a run. What do you think?"

Note: For reproducibility, the complete prompt for Simulated Multi-turn Sessions generation is omitted here for brevity. The underlying methodology involves conditioning the language model on initial character profiles and persistent memory states to synthesize dialogues featuring opinion evolution and affective dynamics.

A.3.3 Data Validation

To ensure the quality of the generated data, we employed a combination of manual spot checks and automated script verification. The spot checks focused on assessing sentiment plausibility, memory logical consistency, and dialogue fluency. Automated scripts were used to check data format compliance and the integrity of basic statistical properties.

A.4 System Parameters

Table 3: Configuration Settings

Parameter	Value
High Entropy Threshold	1.4
Low Entropy Threshold	0.8
Range of S	$[0, 3]$
Retrieval Top-K	5