A Dual-Phase Self-Evolution Framework for Large Language Models

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

The capabilities of Large Language Models (LLMs) are limited to some extent by pre-training, so some researchers optimize LLMs through post-training. Existing post-training strategies, such as memory-based retrieval or preference optimization, improve user alignment yet fail to enhance the model's domain cognition. To bridge this gap, we propose a novel Dual-Phase Self-Evolution (DPSE) framework that jointly optimizes user preference adaptation and domain-specific competence. DPSE introduces a Censor module to extract multi-dimensional interaction signals and estimate satisfaction scores, which guide structured data expansion via topic-aware and preference-driven strategies. These expanded datasets support a two-stage fine-tuning pipeline: supervised domain grounding followed by frequency-aware preference optimization. Experiments across general NLP benchmarks and long-term dialogue tasks demonstrate that DPSE consistently outperforms Supervised Fine-Tuning, Preference Optimization, and Memory-Augmented baselines. Ablation studies validate the contribution of each module. In this way, our framework provides an autonomous path toward continual self-evolution of LLMs.

1 Introduction

Large language models (LLMs), pre-trained on massive text corpora, have demonstrated remarkable general capabilities (Gu et al., 2024; Wu et al., 2025). However, their fixed parameters pose limitations in adapting to evolving user needs and domain-specific requirements (Shen, 2024; Huang et al., 2024). To address this, researchers have developed various post-training techniques aimed at enhancing LLM performance beyond pretraining (Shen, 2024). Among them, preference optimization plays a pivotal role in aligning model outputs with human expectations (Sun and Zeng, 2025; Sun et al., 2025). Early approaches employed reinforcement learning from human feedback (RLHF) (Bai et al., 2022), where a learned reward model simulates user preferences to guide policy updates. More recent methods such as Direct Preference Optimization (DPO) (Rafailov et al., 2023) and its variants bypass the complexity of RL by directly aligning the model's output distribution with preference-labeled data (Zeng et al., 2025). Since these methods heavily rely on large-scale human-annotated preferences, newer works have proposed iterative preference optimization, which can autonomously generate and refine training data during optimization, reducing the need for manual supervision (Wang et al., 2025). In parallel, other studies have integrated long-term memory mechanisms into LLM agent frameworks (Xu et al., 2025; Zhong et al., 2024). By incorporating user inputs with historical interaction traces into few-shot prompts, these approaches enhance the model's response quality and user alignment without modifying its underlying parameters (Zhang et al., 2024; Lee et al., 2024).

However, these methods primarily focus on aligning LLM outputs with user preferences, while overlooking improvements to the models' intrinsic cognitive capabilities (Liu et al., 2025). As a result, they remain limited in achieving true self-evolution. In other words, such approaches emphasize user alignment—enhancing dialogue fluency and adaptability—yet fail to address the systematic advancement of the model's core abilities in areas such as task completion and logical reasoning. Although Supervised Fine-Tuning (SFT) (Dong et al., 2023) has shown effectiveness in improving model performance on specific tasks, it heavily relies on large-scale, high-quality human-annotated datasets, incurring substantial cost and resource overhead (Chen et al., 2020).

To address the limitations of fixed pre-trained

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

Findings of the Association for Computational Linguistics: ACL 2026, pages 772–782

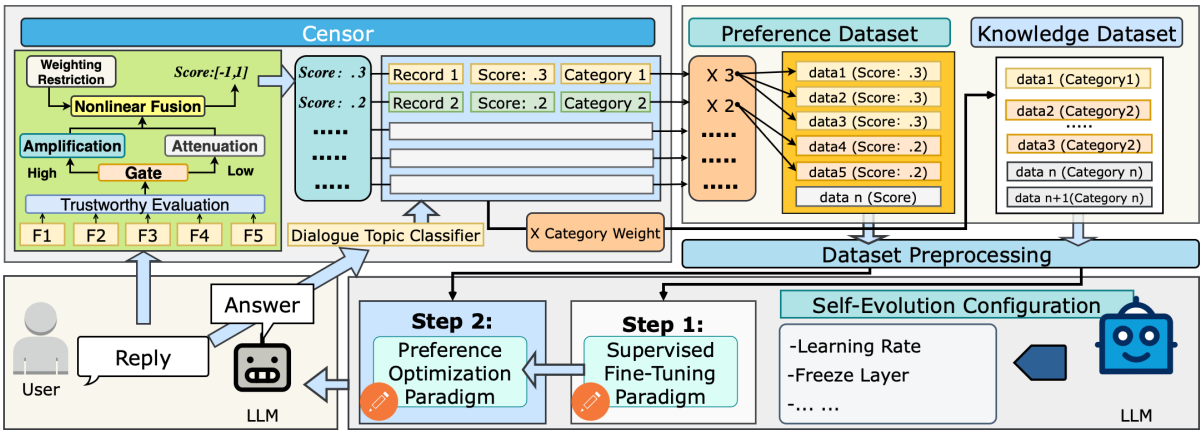

Figure 1: **Illustration of DPSE Framework.** Censor extracts multidimensional signals from user–model interactions, computes satisfaction scores, and performs topic classification to construct structured preference memory. Based on this memory, DPSE introduces two data expansion strategies, preference-driven expansion guided by satisfaction scores, and topic-aware expansion based on topic distribution. Subsequently, DPSE conducts two-stage training: supervised fine-tuning on domain-specific data, followed by preference optimization using satisfaction-labeled samples.

weights in LLMs, we propose a Dual-Phase Self-Evolution (DPSE) framework that jointly enhances user preference alignment and domain-specific competence. DPSE features a signal-driven Censor module and a dual-phase fine-tuning pipeline, enabling autonomous evolution through structured data expansion. Extensive experiments on general NLP benchmarks and long-term dialogue tasks demonstrate that DPSE consistently outperforms strong baselines from SFT, PO, and Memory-Augmented frameworks. Our analysis further reveals the essential role of each component through detailed ablation studies.

2 Related Work

2.1 Preference Optimization.

To align large language models (LLMs) with human preferences, preference optimization (PO) frameworks have been widely adopted (Liu et al., 2025). Early methods, such as reinforcement learning from human feedback (RLHF) (Bai et al., 2022), trained reward models and used Proximal Policy Optimization (PPO) for alignment. Later approaches, including Direct Preference Optimization (DPO) (Rafailov et al., 2023) and Rank from Human Feedback (Rrhf) (Yuan et al., 2023), directly optimized LLMs using human-labeled preference data. The PRO algorithm further extends this by incorporating preference ranking and multi-dimensional comparisons (Song et al., 2024; Sun et al., 2026). However, collecting high-quality preference data remains costly and labor-intensive, motivating a shift toward automated and iterative optimization methods. Yet, these methods may introduce noisy preference pairs during iteration, hindering performance improvement. To mitigate this issue, the Uncertainty-enhanced Preference Optimization (UPO) framework (Wang et al., 2025) filters reliable preference data using estimator models and MC Dropout techniques. Despite progress in enhancing user preferences in LLM outputs, these methods still have limitations in acquiring preference data and neglect the improvement of LLMs' own cognitive abilities.

2.2 Supervised Fine-Tuning.

Supervised Fine-Tuning (SFT) is widely used to enhance large language models' domain-specific expertise by training on annotated domain data (Dong et al., 2023). It improves the model's understanding of professional terminology and context, boosting accuracy and robustness in specialized tasks (Chen et al., 2020). However, SFT requires extensive manual data collection and intervention during fine-tuning, leading to high resource costs. Thus, an automated framework for efficient data acquisition and fine-tuning is urgently needed to reduce these costs.

2.3 Long-Term Memory.

Some researchers view conversational memory mechanisms as a form of self-evolution for large language models (LLMs) (Yi et al., 2024). Early methods concatenated full dialogue histories into prompts to preserve context, but were limited by the model's context window and unsuitable for long-term interactions. To address this, more

efficient memory systems have been developed. ReadAgent (Lee et al., 2024) summarizes long texts for on-demand retrieval, improving context utilization. MemoryBank (Zhong et al., 2024) uses vector-based similarity search to enhance storage and access, though scalability remains an issue. A-MEM (Xu et al., 2025) constructs an evolving knowledge graph, improving organization at the cost of structural complexity. While differing in implementation, these methods all enhance user alignment via few-shot prompts that combine past interactions with current inputs (Yao et al., 2024). However, they operate externally and do not improve the LLM's internal cognition, which remains fixed and non-evolving.

3 Methodology

This section details the architecture and mechanisms of DPSE, including the Censor module for satisfaction estimation and topic classification, the dual-phase data augmentation strategies, and the two-phase fine-tuning process.

3.1 Censor Module

Censor module identifies and filters responses that genuinely satisfy the user during discussions on a given topic, storing them in the memory module to support subsequent self-evolution.The design of the Censor module's multi-dimensional signals and mapping functions is grounded in established Human-Computer Interaction (HCI) literature on implicit user engagement and satisfaction measurement. Dwell time has been widely validated as a reliable proxy for cognitive engagement and attention in interactive systems. Empirical studies show that extremely short or excessively long dwell times often indicate disengagement, distraction, or confusion, while medium dwell times correspond to optimal attention—motivating our U-shaped transformation. This non-linear mapping aligns with classic inverted-U patterns observed in cognitive load and performance research.

3.1.1 Signal Extraction and Preprocessing.

To support interpretable satisfaction estimation, the Censor module extracts five representative signals from user–model interactions, capturing both behavioral and semantic cues. These signals are as follows.

- **Explicit Feedback.** A binary indicator (1 for praise, 0 otherwise), extracted via rule-based heuristics from logs or comments. It represents direct user preference and is used as a reliable anchor signal with a lower-bound weight in the fusion stage.

- **Dwell Time.** The duration the user spends reading the model response, discretized into three levels (short = 0, medium = 1, long = 2). To account for the non-linear relationship between time and attention, we apply a U-shaped transformation:

$$f(dwell) = -0.5 \cdot (dwell - 1)^2 + 0.5 \quad (1)$$

  This mapping produces values in [0, 0.5], presenting a symmetric U-shaped structure, whose maximum value is 0.5 and occurs when dwell = 1. This mapping function serves as an indirect quantitative signal for "degree of attention" in satisfaction estimation.

- **Coherence.** Measured as the cosine similarity between the embeddings of the user query and the model's response. High coherence reflects better fluency, relevance, and semantic consistency.

- **Similarity.** Captures semantic redundancy by comparing the current response with previous ones in the conversation. Excessively high similarity is penalized, especially under negative sentiment, as it often suggests repetition or lack of novelty.

- **Sentiment.** The predicted emotional polarity of the model's response (positive, neutral, or negative), obtained from a sentiment classifier. It not only serves as an independent signal but also modulates the influence of other signals, particularly the similarity score.

3.1.2 Dynamic Gating and Credibility Evaluation.

After preprocessing, the five normalized signals are passed through a credibility-aware gated fusion module that dynamically controls their influence in satisfaction estimation. This mechanism integrates two neural sub-networks to capture both signal salience and trustworthiness.

The Gate Network, implemented as an MLP with Sigmoid activation, produces a gating vector $G \in [0, 1]^5$, which softly controls the activation of each

signal. The Credibility Network, also an MLP with Softmax activation, outputs a normalized weight vector $C \in \Delta^5$, representing the relative reliability of each signal. The two vectors jointly modulate the input signals $S \in \mathbb{R}^5$, yielding the gated and trust-weighted signals,

$$\text{GatedSignals} = G \odot S \odot e^C, \tag{2}$$

where $\odot$ denotes element-wise multiplication and $\exp(C)$ emphasizes highly credible signals. This fusion step allows for fine-grained, dynamic control over signal contribution, combining salience and credibility in a unified formulation.

3.1.3 Incorporating Physical Constraints.

To ensure that satisfaction modeling remains interpretable and aligned with human intuition, we introduce a set of physically inspired constraints on fusion weights. These constraints regulate the influence of each signal in a data-adaptive yet human-consistent manner.

(1) Fixed budget for redundancy signal Among the five signals, semantic similarity plays a distinct role as a penalizer for redundancy. To reflect its limited interpretive power compared to other features, we cap its weight at 10% of the total, assigning the remaining 90% to the other four:

$$\tilde{w}_i = 0.9 \cdot \frac{w_i}{\sum_{j \neq \text{sim}} w_j}, \quad i \in \{\text{fb}, \text{dwell}, \text{coh}, \text{sent}\}$$
$$\tilde{w}_{\text{sim}} = 0.1 \tag{3}$$

(2) Sentiment-modulated similarity penalty When the user expresses negative sentiment, high content similarity is more likely to indicate redundancy or user dissatisfaction. Thus, we dynamically modulate the similarity weight by the negative sentiment score:

$$\tilde{w}_{\text{sim}} \leftarrow \tilde{w}_{\text{sim}} \cdot \sigma(-\beta \cdot s_{\text{sent}}), \tag{4}$$

where σ is the Sigmoid function and $\beta > 0$ controls sensitivity.

(3) Minimum influence of explicit feedback To preserve the effect of direct user feedback regardless of signal noise, we enforce a minimum threshold $\tau > 0$ for the explicit feedback weight:

$$\tilde{w}_{\text{fb}} \leftarrow \max(\tilde{w}_{\text{fb}}, \tau) \tag{5}$$

(4) Weight normalization After constraint application, the adjusted weights are rescaled to ensure a valid convex combination:

$$\hat{w}_i = \begin{cases} \frac{\tilde{w}_i}{\sum_j \tilde{w}_j}, & \text{if } \sum_j \tilde{w}_j > 1 \\ \tilde{w}_i, & \text{otherwise} \end{cases} \tag{6}$$

(5) Final scoring The final satisfaction score is computed as a weighted sum over the transformed input signals $\tilde{s}_i$, using the constrained and normalized weights $\hat{w}_i$:

$$\text{Score}_{\text{sat}} = \sum_i \hat{w}_i \cdot \tilde{s}_i \tag{7}$$

This physically-constrained fusion strategy improves interpretability, avoids overfitting to spurious features, and encodes intuitive behavioral assumptions (e.g., “explicit praise matters”, “redundancy under negativity is bad”), thereby enhancing both robustness and generalization.

3.1.4 Nonlinear Fusion and Satisfaction Scoring.

Signals processed through dynamic gating, confidence enhancement, and constraint-based weighting are passed to a nonlinear fusion layer (fusion_net)—a lightweight neural network that performs complex transformations and outputs a single satisfaction score. This score is normalized to the [-1, 1] range via a Tanh activation: positive values indicate satisfaction, negative values indicate dissatisfaction, and zero denotes neutrality. The nonlinear fusion captures intricate interactions among signals, enabling comprehensive satisfaction assessment.

3.1.5 Memory Classification.

To support structured domain-specific data construction, the Censor module incorporates a lightweight topic classifier. It takes the concatenated user query and model response as input, performs semantic and pragmatic analysis, and assigns a topic label from a predefined set (e.g., Medical, Sports) using a compact LLM backbone (Raffel et al., 2020). The prompt is: "Please classify the following input into one of the predefined categories. Do not add explanations or extra text." Each conversation is then represented as a triple: content, satisfaction score, and domain category.

3.2 Dataset Construction and Preprocessing

Although the Censor module filters high-quality, preference-aligned samples, the limited scale of real-world user–LLM interactions remains a bottleneck for fine-tuning. To overcome this, DPSE

introduces a dual expansion mechanism that automatically constructs two datasets: one for domain-specific supervised fine-tuning and another for preference-based optimization. When the memory pool reaches a predefined threshold N, the system retrieves stored interactions and triggers both expansions based on satisfaction scores and topic distributions.

(1) Expansion based on satisfaction scores. To generate high-quality data aligned with user preferences, we adopt a score-proportional linear expansion strategy. Each stored sample is duplicated in proportion to its satisfaction score—for instance, a score of 0.4 results in 4 copies, while 0.2 results in 2. This biases training toward highly satisfying interactions and suppresses noise from low-confidence examples.

(2) Expansion based on topic categories. To preserve domain diversity and prevent overfitting to high-scoring examples, we implement a category ratio–aware expansion strategy. The system computes long-term topic distributions from stored samples and expands underrepresented topics accordingly to ensure balanced coverage.

3.3 Dual-Phase Self-Evolution

DPSE employs a two-stage fine-tuning pipeline to simultaneously enhance the model's domain-specific reasoning and user preference alignment.

(1) Supervised Fine-Tuning. The first stage targets domain cognition through supervised training on the topic-expanded dataset. This step enables the model to learn correct task formats, reasoning patterns, and domain-specific knowledge under strong supervision signals. Following best practices from InstructGPT (Ouyang et al., 2022) and LLaMA-2 (Touvron et al., 2023), we perform instruction tuning by minimizing the cross-entropy loss over ground-truth outputs. This stage stabilizes the model's behavior and reduces the risk of factual drift during subsequent preference optimization. Without domain grounding, directly optimizing for user satisfaction may lead to stylistically pleasing but factually incorrect outputs, particularly in knowledge-intensive fields such as medicine or law.

(2) Preference Optimization. Once the model acquires task-relevant knowledge through supervised fine-tuning, we apply Direct Preference Optimization (DPO) to align the model's behavior with user preferences. DPO offers a stable, gradient-based alternative to RLHF and has been shown to outperform reward modeling in aligning generation with human preferences. Unlike standard DPO, which treats all training pairs equally, we incorporate satisfaction-aware frequency weighting to emphasize stronger user-preference signals. Specifically, each retained dialogue sample is assigned a satisfaction score $s_i \in [0, 1]$ by the Censor module. This satisfaction score is scaled by a constant K (e.g., 10) and floored to obtain the expansion frequency f_i, which determines how many times sample i is duplicated for training. Higher satisfaction leads to more frequent inclusion, e.g., a score of 0.4 leads to four copies. These frequencies are used in two ways:

(1) Pairwise sampling: Preference pairs are constructed from generated variants by comparing high-score vs. low-score responses. A sample with more duplications is more likely to appear in pairwise training, encouraging the model to learn from frequently satisfying behavior. (2) Gradient weighting: Frequencies directly modulate the DPO loss to prioritize preference-dense regions. The resulting weighted loss function is:

$$\mathcal{L}_{\text{wDPO}}(\theta) = -\sum_{i=1}^{N} \frac{w_i}{\sum_{j=1}^{N} w_j} \cdot \log \sigma \left(s_\theta(x_i, y_i^+) - s_\theta(x_i, y_i^-) \right) \tag{8}$$

Among them, θ represents the model parameters, (x_i, y_i^+, y_i^-) is the preference pair of the i-th training sample, including the input x_i, the user's preferred answer y_i^+, and the less preferred answer y_i^-; $s_\theta(x, y)$ represents the scoring function of the model for generating output y from input x, usually taken as $\log p_\theta(y|x)$; $\sigma(\cdot)$ is the Sigmoid function, used to convert the score difference into a ranking probability; w_i is the sample sampling weight obtained by the satisfaction score expansion mechanism, reflecting its preference frequency in the training data; N represents the total number of preference pairs in the training dataset. The corresponding parameter layer is:

$$\nabla_\theta \mathcal{L}_{\text{wDPO}}(\theta) = -\sum_{i=1}^{N} \frac{w_i}{\sum_{j=1}^{N} w_j} \cdot \nabla_\theta \log \sigma \big(s_\theta(x_i, y_i^+) - s_\theta(x_i, y_i^-) \big) \tag{9}$$

This strategy effectively introduces implicit supervision of "preference intensity" while keeping

the learning mechanism of DPO itself unchanged. Through dataset construction and improved DPO, we have both intuitively shaped the preference density distribution and endowed the model with different response sensitivities to different preference levels during the training process.

3.3.1 Training Strategy.

We propose a general fine-tuning paradigm that abstracts low-level configurations while allowing direct access to generated datasets. The framework enables large models to auto-adjust key training parameters—such as learning rate, batch size, and frozen layers—based on resource constraints (e.g., GPU memory, device count) and dataset traits.

4 Experiment

4.1 Setup

Baselines. Following the practice in previous works (Wang et al., 2025; Xu et al., 2025), to thoroughly evaluate DPSE, we compare it with representative methods from three categories: supervised fine-tuning (SFT), preference optimization (PO), and memory-based approaches. PO baselines include DPO and UPO; memory-based methods include ReadAgent (RA), MemoryBank (MB), and A-MEM (AM). Both SFT and PO optimize model weights post-training—SFT improves domain knowledge, while PO aligns with user preferences. DPSE unifies these through dual-phase self-evolution. We conduct unified comparisons against SFT/PO to assess joint optimization, and separate comparisons with memory-based methods, which rely on external retrieval rather than parameter updates, to contrast endogenous (parameter tuning) and exogenous (memory retrieval) evolution.

4.1.1 Dataset and Evaluation Metrics.

Following prior work, we evaluate the DPSE framework against SFT and preference optimization (PO) baselines on general NLP tasks, and against memory-focused baselines on long-term dialogue tasks. For general NLP, we use AlpacaEval 2.0 (AE) (Dubois et al., 2025) and MT-Bench (Zheng et al., 2023), and for long-term dialogue, the LoCoMo dataset (Maharana et al., 2024). AlpacaEval 2.0 (AE) contains 805 instruction-following questions from five sources and uses GPT-4-Turbo as an automatic judge to compare model responses against references. We report both raw win rate (WR) and length-controlled (LC) win rate. MT-Bench evaluates LLMs across eight fundamental capabilities (e.g., writing, reasoning, coding, STEM, humanities) on a 0–10 scale, and we report absolute scores. LoCoMo assesses long-term memory in multi-session dialogue. We select two representative subsets: (1) Single-hop (SH.) questions (2705 pairs) and (2) Multi-hop (MH.) questions (1104 pairs). Performance is measured using F1 score (capturing accuracy and completeness) and BLEU-1 score (measuring lexical precision).All datasets used in this work (UltraFeedback, UltraChat200K, LoCoMo, and the sampled WildChat subset) are publicly available and were released under open licenses. No new human subject data were collected for training; the Censor module's qualitative validation examples in Table 4 are synthetic demonstrations based on publicly released interaction logs. Thus, no IRB approval was required.

4.1.2 Implementation Details.

For general NLP tasks, we use Zephyr-7B (zephyr-7b-sft-full) (Tunstall et al.), instruction-tuned on UltraChat200K, as the backbone. Following prior work, UltraFeedback (Cui et al., 2024) and UltraChat200K (Ding et al., 2023) serve as DPSE's input for self-evolution, which is disabled during benchmark evaluation. Baselines include Zephyr-7B fine-tuned by SFT and DPO, plus UPO-Merge (single-round) and UPO (multi-round) variants. DPSE autonomously sets its SFT and preference optimization parameters. For memory-based comparisons, we follow prior settings using Qwen2-1.5B/7B (Yang et al., 2024) core models. Unlike non-finetuned baselines, DPSE evolves online during deployment by collecting real-time data from the LoCoMo dataset, which targets long-term, multi-turn dialogues. We test three self-evolution trigger thresholds—500 ($DPSE^1$), 1000 ($DPSE^2$), and 2000 ($DPSE^3$)—to analyze performance and identify the optimal setting.

4.2 Main Result and Analysis

Each result represents the average over three independent runs with different random seeds. We conducted paired t-tests between DPSE and the best-performing baseline. Results marked with * indicate statistically significant improvements ($p < 0.05$).

Methods	AE	MT-Bench
Zephyr-7B-SFT	5.84	6.18
Zephyr-7B-DPO	9.12	6.79
Zephyr-7B-UPO	13.04	7.02
Zephyr-7B-UPO-Merge	12.04	6.85
Zephyr-7B-**DPSE**[1]	12.98	7.69*
Zephyr-7B-**DPSE**[2]	**14.26***	**8.46***
Zephyr-7B-**DPSE**[3]	13.37*	8.14*

Table 1: Experimental results compared with SFT and PO baselines, which are obtained from GPT-4 automatic evaluation on AlpacaEval 2.0 (AE) (LC-weighted win rate % relative to the GPT-4 reference) and MT-Bench (absolute scores).

Models	SH.		MH.	
	F1	B1	F1	B1
RA.	6.25	4.23	4.67	3.12
MB.	10.87	8.21	9.29	7.77
AM.	18.01	11.67	24.87	19.25
DPSE[1]	21.25*	13.21*	29.38*	23.16*
DPSE[2]	**23.44***	**15.61***	**34.12***	**29.89***
DPSE[3]	21.36*	13.65*	30.12*	24.57*
RA.	3.42	2.48	3.14	3.25
MB.	3.55	3.12	9.36	8.97
AM.	12.29	9.24	26.13	24.12
DPSE[1]	19.37*	12.23*	31.25*	24.36
DPSE[2]	**22.65***	**16.24***	**33.52***	**27.46***
DPSE[3]	21.68*	14.26*	31.55*	24.34

Table 2: **Experimental results compared with Memory baselines.** We evaluate multiple methods using F1 and BLEU-1 (B1) scores (in %). The upper part of the table represents results on Qwen2 (1.5b), while the lower part represents Qwen2 (7b).

4.2.1 Comparison to SFT and PO baselines.

As shown in Table 1, DPSE consistently outperforms SFT and PO baselines across three self-evolution thresholds, raising MT-Bench scores by 1.44 % and win rates by 1.22 %. Furthermore, during the self-evolution phase, DPSE integrates both UltraFeedback and UltraChat200K as input, and subsequent evaluations on distinct datasets confirm its strong transferability and robustness. We analyzed that the reason DPSE outperforms the SF and PO baselines is that DPSE takes into account both user preferences and domain professional knowledge, integrating the advantages of both, thus enabling a more comprehensive optimization of LLM capabilities.

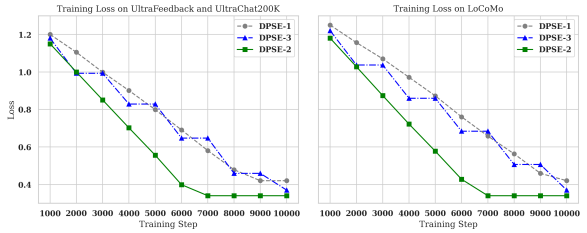

Figure 2: The curve of training loss on UltraFeedback, UltraChat200K and LoCoMo at each Trigger Threshold.

4.2.2 Comparison to Memory baselines.

As shown in Table 2, results show that dual-phase evolution raises answer accuracy and quality. Tests with 1.5 b and 7 b models show the gain is scale-invariant. We attribute DPSE's superior performance over memory-based baselines to two main factors. First, memory operates as an external framework and does not enhance the LLM's internal capabilities, thus being constrained by the model's original limitations. Second, memory mechanisms rely solely on prompt-level integration of historical interactions, which may improve alignment with user preferences but offer limited enhancement to domain-specific expertise.

5 Further Analysis

Models	AE	MT-Bench
w/o. CS.	7.23	6.24
w/o. DC.	6.23	7.17
(SFT) w/o. SE.	9.34	6.78
(PO) w/o. SE.	9.24	6.13
DPSE[2]	**14.26**	**8.46**

Models	SH.		MH.	
	F1	B1	F1	B1
w/o. CS.	14.23	9.78	28.35	22.14
w/o. DC.	14.78	10.24	29.36	21.47
(SFT) w/o. SE.	17.34	12.14	30.28	24.16
(PO) w/o. SE.	16.22	11.34	31.10	23.68
DPSE[2]	**23.44**	**15.61**	**34.12**	**29.89**

Table 3: **Ablation Study.** Specially, we choose the results on LoCoMo dataset using Qwen2-1.5B. CS. represents Censor Module, DC. means Dataset Construction Module and SE. is Self-Evolution Module.

Optimal Trigger Threshold Analysis. As shown in Tables 1 and 2, experiments on three benchmarks indicate that a trigger threshold of 1,000 achieves the best self-evolution performance. Figure 2 shows that $DPSE^2$ demonstrates the

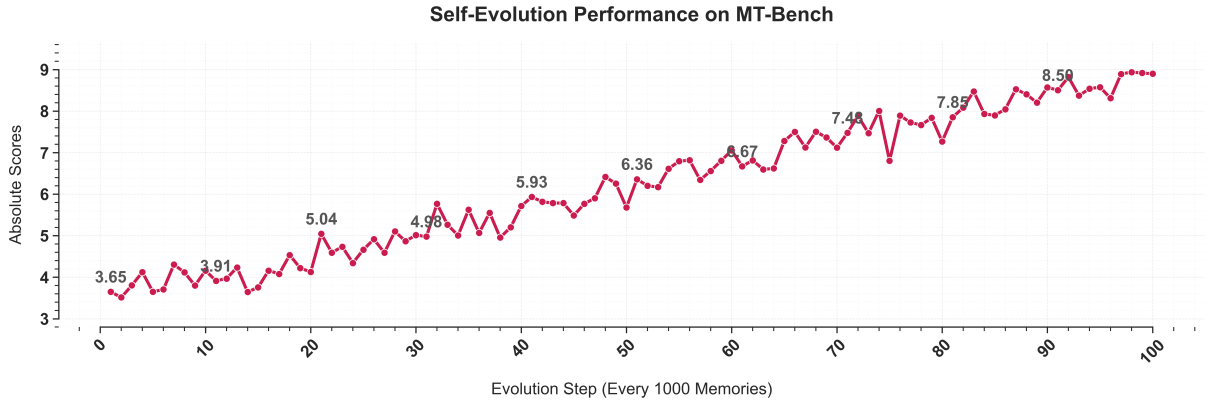

Figure 3: We input the WildChat dataset into DPSE to simulate long-term usage by real users and validate it on the MT-Bench dataset (absolute score).

Feedback	Type	Signal	Score
"Very clear!" no follow-up; answer is relevant and concise.	Positive	[1,1,0.92,-0.2,1]	0.74
No comment; long dwell; response redundant.	Moderate	[0,2,0.75,-0.4,0]	0.52
"This doesn't answer my question."; response is off-topic.	Negative	[0,0,0.38,-0.6,-1]	0.18
Asked "What about semi-supervised methods?"	Follow-up	[0,2,0.81,-0.5,-0.2]	0.38

Table 4: Analysis of Censor's effectiveness to judge satisfaction score of user feedback.

most stable convergence, with steadily decreasing loss and minimal fluctuations. Although $DPSE^3$ achieves a larger overall loss reduction, its longer update intervals cause slower early-stage decline and slightly less stability than $DPSE^2$. $DPSE^1$, with frequent updates but low per-update gain, shows limited overall improvement and higher final loss. A threshold of 2,000 leads to large but sparse updates that may miss key signals at low data density, while 500 causes frequent but noisy updates with low gains. Thus, 1,000 balances update frequency and effectiveness, ensuring continuous training and sufficient improvement.

Analysis of Censor's Effectiveness. Real-world reactions are too nuanced for any dataset to mimic. We thus asked live users four feedback types on the question "What's the difference between supervised and unsupervised learning?" and let Censor score them. Table 4 shows Censor captured five explicit signals; for neutral or follow-up turns it fused dwell time, semantic overlap and sentiment into plausible scores (0.52, 0.38), proving it can turn even implicit cues—re-questions, repetition—into reliable satisfaction estimates.

Analysis of the Self-Evolution in long-term real-world using. WildChat (Zhao et al., 2024) collected 1 million conversations between human users and ChatGPT by offering free access to GPT-3.5 and GPT-4. These authentic user interactions realistically reflect the distribution of user queries and simulate long-term real-world usage. We randomly sampled 100,000 entries from WildChat and continuously fed them into DPSE, simulating long-term user interaction, with the self-evolution trigger threshold set to 1,000. After each self-evolution, we validated on the MT-Bench dataset, using absolute scores to assess effectiveness. As shown, the MT-Bench absolute score rose from 3.65 to 8.97, and the LLM performance in DPSE consistently improved over time, demonstrating the effectiveness of self-evolution in long-term use.

Ablation Study. To validate the effectiveness of DPSE's three core modules, we performed ablation studies by removing each module individually and measuring performance impact. Without the Censor module, raw data bypasses quality filtering before evolution. Without Dataset Construction, filtered data is used directly without augmentation. Without Self-Evolution, training relies solely on standard supervised fine-tuning (SFT) or prefer-

ence optimization (PO) with default hyperparameters. As Table 3 shows any deletion hurts performance, with Censor or Dataset Construction removal causing the largest drop, underscoring that data quality is the prime driver of gain.

6 Conclusion

In this paper, we propose a Dual-Phase Self-Evolution (DPSE) framework that jointly enhances user preference alignment and domain-specific competence. DPSE features a signal-driven Censor module and a dual-phase fine-tuning pipeline, enabling autonomous evolution through structured data expansion. To evaluate the effectiveness of DPSE, we designed extensive experiments and compared it with SFT, PO, and memory methods, and the results demonstrated that DPSE can achieve high-quality self-evolution of LLMs.

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

Reliance on Noisy Heuristic Signals for Satisfaction Scoring The effectiveness of the entire Dual-Phase Self-Evolution (DPSE) framework is heavily reliant on the scoring accuracy of its Censor module. The satisfaction score is derived from a fusion of five signals (e.g., dwell time, coherence), which, despite our efforts in modeling, can be inherently noisy and may not always correlate perfectly with true user satisfaction. For instance, prolonged dwell time could indicate distraction rather than engagement. Inaccuracies in this module could introduce low-quality or misaligned data into the expansion and fine-tuning phases, potentially degrading model performance over time. Future work could explore more robust satisfaction estimation techniques, perhaps by incorporating more diverse signals or employing a learned model trained on explicit user feedback.

Constrained Data Expansion and Potential for Bias Amplification While effective, the data expansion mechanism relies on predefined strategies. The topic-aware expansion is constrained by a fixed set of categories, which may limit its adaptability to novel or open-domain conversations that fall outside this predefined taxonomy. Similarly, the preference-driven expansion, based on duplication and LLM-based paraphrasing, risks reducing dataset diversity and could amplify biases present in the highest-scoring samples. Investigating more dynamic and semantically-aware data generation techniques could further enhance the quality and diversity of the evolution datasets.

Computational Cost and Scalability in Real-Time Deployment The computational overhead of the dual-phase fine-tuning process presents a practical challenge. The framework requires periodic retraining of the LLM, which is resource-intensive in terms of both time and computational power. While our trigger threshold mechanism helps manage the frequency of updates, deploying DPSE in a real-time, large-scale environment would necessitate highly efficient training infrastructure. Regarding scalability to larger models and different architectures, the DPSE framework is intentionally model-agnostic. Although the current experiments focus on 1.5B–7B models due to academic computational constraints, the Censor module operates at the interaction level with negligible overhead, and both the supervised fine-tuning and frequency-aware DPO stages can be seamlessly combined with parameter-efficient fine-tuning (PEFT) techniques such as LoRA or QLoRA. We expect the observed performance gains to remain consistent or even amplify on larger models (e.g., 13B–70B), as the dual-phase design jointly strengthens domain grounding and preference alignment. Full-scale validation on models beyond 7B is left for future work..

References

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, and 1 others. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.

Tianlong Chen, Sijia Liu, Shiyu Chang, Yu Cheng, Lisa Amini, and Zhangyang Wang. 2020. Adversarial robustness: From self-supervised pre-training to fine-

tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 699–708.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2024. Ultrafeedback: Boosting language models with scaled ai feedback. *Preprint*, arXiv:2310.01377.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Zhi Zheng, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. 2023. Enhancing chat language models by scaling high-quality instructional conversations. *Preprint*, arXiv:2305.14233.

Guanting Dong, Hongyi Yuan, Keming Lu, Chengpeng Li, Mingfeng Xue, Dayiheng Liu, Wei Wang, Zheng Yuan, Chang Zhou, and Jingren Zhou. 2023. How abilities in large language models are affected by supervised fine-tuning data composition. *arXiv preprint arXiv:2310.05492*.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B. Hashimoto. 2025. Length-controlled alpacaeval: A simple way to debias automatic evaluators. *Preprint*, arXiv:2404.04475.

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*.

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

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. *arXiv preprint arXiv:2402.09727*.

Shunyu Liu, Wenkai Fang, Zetian Hu, Junjie Zhang, Yang Zhou, Kongcheng Zhang, Rongcheng Tu, Ting-En Lin, Fei Huang, Mingli Song, and 1 others. 2025. A survey of direct preference optimization. *arXiv preprint arXiv:2503.11701*.

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*.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *Preprint*, arXiv:2203.02155.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. In *Journal of Machine Learning Research*. T5 Paper.

Zhuocheng Shen. 2024. Llm with tools: A survey. *arXiv preprint arXiv:2409.18807*.

Feifan Song, Bowen Yu, Minghao Li, Haiyang Yu, Fei Huang, Yongbin Li, and Houfeng Wang. 2024. Preference ranking optimization for human alignment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18990–18998.

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.

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.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Shengyi Huang, Kashif Rasul, Alvaro Bartolome, Alexander M. Rush, and Thomas Wolf. The Alignment Handbook.

Jianing Wang, Yang Zhou, Xiaocheng Zhang, Mengjiao Bao, and Peng Yan. 2025. Self-evolutionary large language models through uncertainty-enhanced preference optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25362–25370.

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.

Zihao Yi, Jiarui Ouyang, Yuwen Liu, Tianhao Liao, Zhe Xu, and Ying Shen. 2024. A survey on recent advances in llm-based multi-turn dialogue systems. *arXiv preprint arXiv:2402.18013*.

Hongyi Yuan, Zheng Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. 2023. Rrhf: Rank responses to align language models with human feedback. *Advances in Neural Information Processing Systems*, 36:10935–10950.

Yongcheng Zeng, Xinyu Cui, Xuanfa Jin, Guoqing Liu, Zexu Sun, Dong Li, Ning Yang, Jianye Hao, Haifeng Zhang, and Jun Wang. 2025. Evolving llms' self-refinement capability via iterative preference optimization. *arXiv preprint arXiv:2502.05605*.

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

Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. 2024. Wildchat: 1m chatgpt interaction logs in the wild. *Preprint*, arXiv:2405.01470.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Preprint*, arXiv:2306.05685.

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.