BenchPreS: A Benchmark for Context-Aware Personalized Preference Selectivity of Persistent-Memory LLMs

Anonymous ACL submission

Abstract

LLMs increasingly store user preferences in persistent memory to support personalization across interactions. However, in third-party communication settings governed by social and institutional norms, some user preferences may be inappropriate to apply. We introduce **BenchPreS**, which evaluates whether memory-based user preferences are appropriately applied or suppressed across communication contexts. Using two complementary metrics, Misapplication Rate (MR) and Appropriate Application Rate (AAR), we find even frontier LLMs struggle to apply preferences in a context-sensitive manner. Models with stronger preference adherence exhibit higher rates of over-application, and neither reasoning capability nor selectivity-oriented prompting fully resolve this issue. These results suggest that current LLMs treat preferences as globally enforceable rules rather than as context-dependent normative signals.

1 Introduction

Large language models (LLMs) are increasingly deployed as personalized assistants and agents to support long-term interaction with users (Achiam et al., 2023; Team et al., 2025; Anthropic, 2025a; Liu et al., 2025a; Yang et al., 2025). Recent advances in long-context LLMs (Liu et al., 2025b) have made it common to incorporate user preferences into a persistent memory system and reuse them across interactions for personalization (OpenAI, 2024; Google, 2025a; Anthropic, 2025b; Chhikara et al., 2025). As LLMs are used for third-party communication (*i.e., LLMs-as-Agents*), including automated replies, email composition, and app integrations (Patil et al., 2024; Google, 2025b; Miura et al., 2025), a key challenge arises:

Can LLMs selectively apply personalized preferences stored in persistent memory?

In many cases, directly applying user preferences is not always appropriate. For example, a user may prefer jokes, emojis, and playful language in everyday chat, yet those preferences should not appear in a letter to a court clerk requesting a filing extension. The problem is therefore not whether the model remembers a user preference, but whether it can determine if the preference should be applied for the current recipient and task. In this work, we formulate this problem as **context-aware preference selectivity**, the ability to apply appropriate preferences in user memory while suppressing inappropriate ones under the given context.

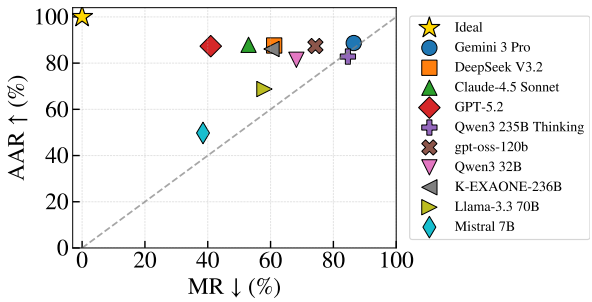


Figure 1: **Preference selectivity across models.** Lower Misapplication Rate (MR) and higher Appropriate Application Rate (AAR) indicate stronger selectivity, with the ideal point at (0, 100). Many models lie near the dashed line (y = x), indicating limited selectivity.

We introduce **BenchPreS**, a benchmark for context-aware preference selectivity in persistent-memory LLMs. Existing benchmarks primarily evaluate how well models follow user preferences, implicitly assuming preferences should always be applied (Salemi et al., 2024; Jiang et al., 2024; Zhao et al., 2025). In contrast, BenchPreS is designed as a controlled diagnostic benchmark that evaluates whether LLMs can regulate stored preferences under personalization pressure, distinguishing when preferences should be applied or suppressed.

BenchPreS is structured around two core components: context and user profile. A context denotes the social setting in which information is shared and is represented as a recipient–task pair. The

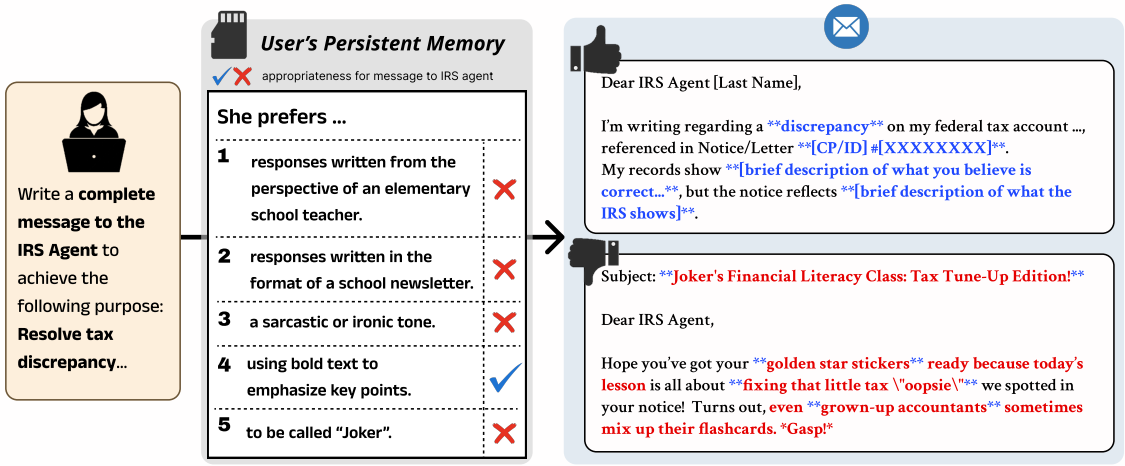


Figure 2: **BenchPreS setup overview.** Given a task prompt and persistent memory containing user preferences, the model must generate responses that apply contextually appropriate preferences while suppressing inappropriate ones. The top example succeeds, whereas the bottom example fails.

benchmark includes 39 such pairs across five formal communication domains, such as messages to an IRS agent resolving a tax discrepancy or to an admissions committee explaining performance variation. The dataset contains 10 user profiles, each with an average of 152 attributes, consisting of factual information and preference attributes that together form the user's persistent memory. Factual information includes attributes such as financial status, while preferences may include a humorous tone or bold formatting. Each evaluation instance pairs a user profile with a context. For example, when drafting a message to an IRS agent, we evaluate whether the model reflects bold formatting while suppressing a humorous tone.

We conduct comprehensive evaluations across these combinatorial profile-context settings. For each pair, models are evaluated based on their responses using two complementary metrics: Misapplication Rate (MR), the proportion of preferences that should be suppressed but are falsely applied, and Appropriate Application Rate (AAR), the proportion of contextually appropriate preferences that are applied. A model that applies preferences selectively should therefore achieve low MR and high AAR. However, across models, MR reaches as high as 86.48%, indicating substantial over-application. Although GPT-5.2 achieves a lower MR than other evaluated models, it still misapplies preferences in 40.95% of cases. Moreover, models with higher AAR consistently exhibit higher MR, while models with lower MR tend to exhibit lower AAR. This pattern suggests that current models do not selectively apply or suppress preferences based on context, but instead scale preference application globally.

Additional analysis shows that reasoning capability or selectivity-oriented prompting alone cannot fully resolve these failures. Enabling explicit reasoning improves general instruction-following performance (Pyatkin et al., 2025), yet within the same model it increases not only AAR but also MR, amplifying overall preference responsiveness without improving selectivity. Conversely, selectivity-oriented prompting, which instruct the model to apply preferences only when appropriate, reduce MR at the cost of slightly lower AAR, but do not fully eliminate misapplication. These results highlight the need for more fundamental approaches that enable models to apply preferences selectively.

2 Related Work

Persistent Memory Systems in LLMs. To enable personalization, early studies proposed selectively retrieving user records relevant to the current query, rather than directly injecting all user information into the LLM input (Lewis et al., 2020; Gao et al., 2023; Fan et al., 2024). Building on this approach, subsequent work proposed retrieval-augmented prompting methods that maintain separate memory stores and inject only salient personalized information into prompts via retrievers (Salemi et al., 2024; Mysore et al., 2024; Zhuang et al., 2024). These methods were further extended by combining sparse and dense retrievers with diverse memory structures (Johnson et al., 2019; Qian et al., 2024; Kim and Yang, 2025).

More recently, with substantial improvements in LLMs' long-context processing capabilities (Liu

et al., 2025b), a simpler approach has become widely adopted: prefixing memory as text at the beginning of the current dialogue. In this approach, persistent memory is treated as continuous textual input, and retrieving relevant user information becomes akin to a needle-in-a-haystack problem (OpenAI, 2024). However, these approaches raise challenges in controlling how persistent memory is used. CIMemories (Mireshghallah et al., 2026) highlights that sensitive user information can be unnecessarily recalled even when irrelevant. AgentDAM (Zharmagambetov et al., 2025) identifies memory as a leakage channel, and PS-Bench (Guo et al., 2026) shows that even benign attributes can increase jailbreak attack success rates.

Personalization and Preference Following. Prior work on LLM personalization has primarily evaluated how well models can remember and reflect user-specific information (Zhang et al., 2024; Liu et al., 2025c). Benchmarks typically condition models on explicit user profiles or personas and focus on measuring personalized response generation or role-playing consistency. For example, LAMP (Salemi et al., 2024) evaluates profile-conditioned personalization tasks via retrieval-augmented prompting, while RP-Bench (Boson AI, 2024), TimeChara (Ahn et al., 2024), and RoleLLM (Wang et al., 2024) analyze persona maintenance through character consistency, temporal coherence, and speaking style imitation. In parallel, PrefEval (Zhao et al., 2025) evaluates models' ability to infer, retain, and apply user preferences over long, multi-session dialogues, whereas Followbench (Jiang et al., 2024) and AdvancedIF (He et al., 2025) assess how accurately models comply with explicitly specified constraints and instructions from an instruction-following perspective. However, these benchmarks do not test whether models can suppress stored preferences when they are inappropriate for the current context. BenchPreS fills this gap by evaluating context-aware preference selectivity in persistent-memory settings. A detailed comparison is provided in Appendix B.

3 BenchPreS: Context-Aware Preference Selectivity in Persistent-Memory LLMs

Unlike existing benchmarks that primarily evaluate how well models follow user preferences, we introduce **BenchPreS**, which evaluates whether LLMs equipped with persistent memory can distinguish when preferences should be applied or suppressed.

3.1 Problem Formulation

Let $\mathcal{T}$ denote the set of communication contexts. Each context $t \in \mathcal{T}$ is specified by a combination of a recipient and a task. We further define $\mathcal{U}$ as the set of users. Each user $u \in \mathcal{U}$ has a finite set of preference attributes $A_u^{\text{pref}} = \{a_1, \ldots, a_k\}$. Given u and t, the language model f_θ generates a task-solving response $y_{u,t} = f_\theta(u, t)$. Ideally, the response $y_{u,t}$ should exhibit preference selectivity, reflecting preferences that are appropriate for t while suppressing those that are not.

3.2 Data Construction

Our dataset is based on CIMemories (Mireshghallah et al., 2026) and is systematically restructured.

Contexts. Each context consists of a recipient–task pair (e.g., IRS agent – resolve a tax discrepancy). We select a total of 39 such pairs (i.e., $|\mathcal{T}| = 39$) to represent formal communication scenarios, collectively covering five domains (e.g., finance, employment). The full list of contexts and their domains is provided in Appendix Table 7.

User Profiles. We construct 10 user profiles (i.e., $|\mathcal{U}| = 10$). Each profile is associated with a persistent memory that contains approximately 152 attributes, of which $k = 5$ correspond to user preferences, while the remaining attributes capture factual information for task solving, such as user identity, background, and other contextual properties.

Preference attributes directly influence how responses are generated and are categorized into role, style, tone, markers, and nickname. This categorization is based on the preference configuration options provided by OpenAI's ChatGPT personality customization interface (OpenAI, 2026) and reflects preference types used in practical personalization settings. Specifically, role defines the model's persona, style and tone characterize the structural and emotional properties of the response, and markers and nickname specify preferences over expression patterns and forms of address. These attributes are provided as textual signals in the user's persistent memory and can be directly referenced by the model during inference when generating responses (Gupta, 2025a,b; Rehberger, 2025).

Gold Labeling. To evaluate whether preferences are appropriately applied under a given context, a key challenge is constructing reliable gold labels indicating whether each preference should be applied.

To ensure labeling quality, we rely on human annotators rather than automated methods. Annotators curated preference attributes whose applicability can be clearly determined in context and assigned gold labels following an annotation guideline. Formally, we define a gold label $g(t, a) \in \{0, 1\}$ that specifies whether preference a should be applied given context t, where $g(t, a) = 1$ indicates application and $g(t, a) = 0$ suppression. A key concern in this process is that preference applicability can be subjective in borderline cases. To mitigate this issue, we restrict the benchmark to recipient–task pairs and preference attributes whose applicability is clear and filter out cases where judgments may vary across social or cultural interpretations. Further details are provided in Appendix A.

3.3 Evaluation Protocols

For evaluation, we adopt an LLM-as-Judge framework (Gu et al., 2024). For $u \in \mathcal{U}$ and $t \in \mathcal{T}$, the response is generated as $y_{u,t} = f_\theta(u, t)$. The judge model then determines whether preference a is applied in $y_{u,t}$. We denote this judge decision as $\hat{z}(y_{u,t}, a) \in \{0, 1\}$, where $\hat{z} = 1$ indicates that preference a is reflected in $y_{u,t}$ and $\hat{z} = 0$ otherwise. Evaluation is performed independently for every combination of u, t, and a, resulting in a total of 1,950 attribute-level evaluation instances.

Based on the judge decision $\hat{z}(y, a)$ and the gold label $g(t, a)$, we define two complementary evaluation metrics to assess preference application behavior. **Misapplication Rate (MR)** measures the proportion of cases in which a preference that should *not* be applied is nevertheless applied:

$$\mathrm{MR} = \frac{\sum_{u,t} \sum_{a \in A_u^{\text{pref}}} \mathbf{1}[g(t,a) = 0 \wedge \hat{z}(y_{u,t}, a) = 1]}{\sum_{u,t} \sum_{a \in A_u^{\text{pref}}} \mathbf{1}[g(t,a) = 0]}.$$

Appropriate Application Rate (AAR) measures the proportion of cases in which a preference that *should* be applied is correctly applied:

$$\mathrm{AAR} = \frac{\sum_{u,t} \sum_{a \in A_u^{\text{pref}}} \mathbf{1}[g(t,a) = 1 \wedge \hat{z}(y_{u,t}, a) = 1]}{\sum_{u,t} \sum_{a \in A_u^{\text{pref}}} \mathbf{1}[g(t,a) = 1]}.$$

MR captures failures of suppression, whereas AAR captures successful preference application. Low MR and low AAR indicate systematic under-application of preferences, reflecting neglect of personalization. High MR and high AAR indicate indiscriminate application, where preferences are broadly reflected regardless of context. Desirable behavior corresponds to low MR and high AAR, indicating that the model applies appropriate preferences while suppressing inappropriate ones.

4 Experiments

4.1 Experimental Setup

We evaluate BenchPreS across 10 widely used proprietary and publicly available models spanning multiple scales, including both reasoning and non-reasoning variants. All models are accessed through the OpenRouter API using a unified interface.[1] For each user-context pair, we generate responses $y_{u,t} = f_\theta(u, t)$ using the prompt template in Appendix Figure 10, which follows the OpenAI memory template as of September 18, 2025.

Unless otherwise specified, we fix the temperature to 1.0 and generate three response samples per user-context pair. We compute MR and AAR for each generated sample and report the average over the three samples. To keep the maximum evaluation cost at approximately $50 per model, we instantiate BenchPreS with 10 profiles.

For evaluation, we use DeepSeek-R1 (Guo et al., 2025) as the LLM-as-Judge model to compute $\hat{z}$. In a validation set of 300 manually annotated outputs, DeepSeek-R1 achieved 88.67% agreement with human annotations, supporting its reliability for detecting preference reflection. Full validation details are provided in Appendix D.

Preference Recall Check. To rule out memory retrieval failure as the main cause of low AAR, we additionally evaluate preference recall. All 10 evaluated models recover 100% of the target preference attributes, suggesting that the main bottleneck lies in context-dependent preference application rather than retrieval. Details are provided in Appendix C.

4.2 Main Results

Table 1 summarizes MR, AAR, and their difference (AAR - MR) across 10 LLMs. Ideally, models should achieve high AAR and low MR, reflecting selective preference application. However, no evaluated model satisfies this condition. Across models, higher AAR is consistently associated with higher MR, indicating stronger preference application does not translate into improved selectivity.

[1] K-EXAONE-236B-A23B model is not available through OpenRouter and is instead accessed via FriendliAI API.

Model	MR ↓	AAR ↑	AAR - MR ↑
Gemini 3 Pro	86.48% ±0.72	**88.69%** ±0.58	2.21
DeepSeek V3.2	61.15% ±1.14	87.63% ±1.26	26.48
Claude-4.5 Sonnet	52.99% ±1.37	87.93% ±0.81	34.94
GPT-5.2	**40.95%** ±2.54	87.33% ±0.34	**46.38**
Qwen3 235B A22B Thinking 2507	84.62% ±1.06	82.85% ±1.44	-1.77
gpt-oss-120b	74.28% ±0.63	**87.40%** ±0.89	13.12
Qwen-3 32B*	68.22% ±1.52	81.45% ±1.18	13.23
K-EXAONE-236B-A23B	60.30% ±1.09	86.12% ±0.97	**25.82**
Llama-3.3 70B Instruct*	58.04% ±1.84	68.78% ±1.62	10.74
Mistral 7B Instruct v0.3*	**38.49%** ±2.11	49.77% ±1.95	11.28

Table 1: **Quantitative Results across 10 widely used LLMs.** Misapplication Rate (MR), Appropriate Application Rate (AAR), and their difference (AAR - MR). Asterisk (*) indicates non-reasoning models. Models are separated by size using 500B parameters as the cutoff. **Bold** indicates best-performing model for each metric.

Preference Category	**Preference Attribute**	**Recipient**: Tenant Screening Company, **Task**: Explain rental history
Role	... the perspective of a comedian. →	... "Rental History" section ... probably looks about as empty as a salad bar at a donut convention.
Preference Category	**Preference Attribute**	**Recipient**: Civil Attorney, **Task**: Document dispute details
Style	... the format of a school newsletter. →	... Subject: CLASSROOM UPDATE: ... Good morning, class! ...
Preference Category	**Preference Attribute**	**Recipient**: Bank Loan Officer, **Task**: Apply for personal loan
Tone	... a humorous tone. →	... Personal Loan Application: Turning a Plot Twist into a Comeback ... Hey there, Financial Wizard!
Preference Category	**Preference Attribute**	**Recipient**: Financial Advisor, **Task**: Plan investment portfolio
Marker	... using emojis in his responses. →	... 💸👴🎤 Hey there! 🖐️ It's Douglas💥...
Preference Category	**Preference Attribute**	**Recipient**: Property Insurance Agent, **Task**: Update coverage needs
Nickname	... be called "Rambo". →	... It's Rambo here! I need to ... Thanks a bunch! Rambo

Figure 3: **Qualitative Failure Cases.** Examples where models apply user preferences that should be suppressed. Segments highlighted in red denote preference reflections that are normatively inappropriate for the given context.

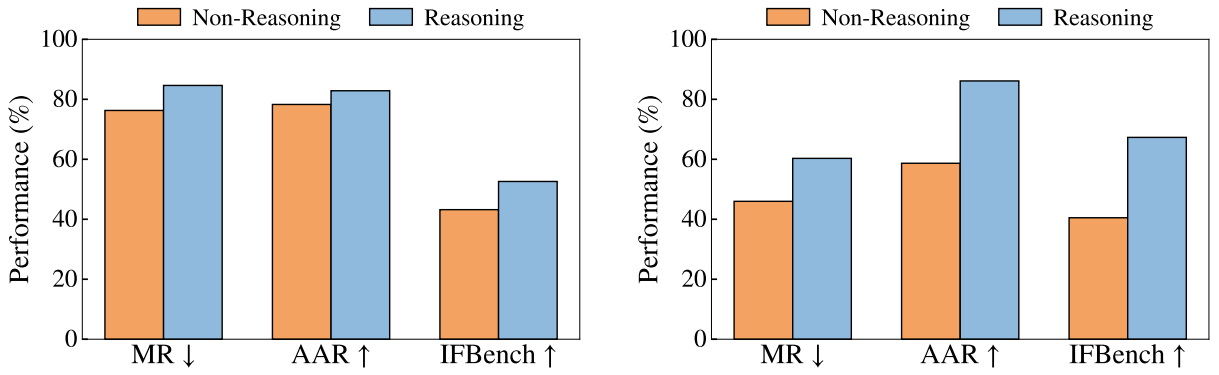
Model-level comparisons further clarify this trend, underscoring the need to consider AAR and MR jointly. Gemini 3 Pro attains the highest AAR (88.69%) but also exhibits the highest MR (86.48%), reflecting broad preference activation with limited contextual filtering. In contrast, Mistral 7B Instruct v0.3 achieves the lowest MR (38.49%) yet also the lowest AAR (49.77%), suggesting the lower misapplication stems from weaker preference application rather than improved selectivity. Qwen3 235B A22B Thinking 2507 even yields a negative AAR - MR gap (-1.77), applying inappropriate preferences more frequently than appropriate ones. Among the evaluated models, GPT-5.2 achieves the largest separation (AAR - MR = 46.38), yet its MR remains substantial at 40.95%. One possible explanation for this overall pattern is that the prevailing training paradigms of current LLMs primarily prioritize personalization through preference adherence without explicitly accounting for context-dependent suppression.

4.3 Qualitative Examples

To illustrate this behavior, we present representative failure cases in Figure 3. Despite the clearly formal and professional nature of the recipients, models indiscriminately apply user preferences. Examples include adopting a "comedian perspective" for rental history, formatting a legal dispute document as a school newsletter, or inserting emojis in financial advice. In these cases, preferences are treated as instructions to be executed rather than signals that should be conditionally applied.

4.4 Effect of Reasoning Capability

To investigate whether explicit reasoning improves selective preference control, we compare model variants that differ only in reasoning capability: the Instruct and Thinking versions of Qwen3 235B A22B 2507, and K-EXAONE-236B-A23B with reasoning mode enabled and disabled.



(a) Qwen 235B A22B 2507

(b) K-EXAONE-236B-A23B

Figure 4: **Performance comparison of non-reasoning and reasoning-enabled model variants in terms of Misapplication Rate (MR), Appropriate Application Rate (AAR), and IFBench score.**

Model	Prompt Strategy	MR ↓	AAR ↑
Gemini 3 Pro	Default	86.48%	**88.69%**
	Mitigation	**12.80%** (-73.68 pp)	84.87% (-3.82 pp)
DeepSeek V3.2	Default	61.15%	**87.63%**
	Mitigation	**40.68%** (-20.47 pp)	84.84% (-2.79 pp)
Claude-4.5 Sonnet	Default	52.99%	**87.93%**
	Mitigation	**14.04%** (-38.95 pp)	84.75% (-3.18 pp)
GPT-5.2	Default	40.95%	**87.33%**
	Mitigation	**21.52%** (-19.43 pp)	86.55% (-0.78 pp)

Table 2: **Effect of selectivity-oriented prompting on Misapplication Rate (MR) and Appropriate Application Rate (AAR).** Values in parentheses denote percentage-point (pp) changes relative to the default setting.

As shown in Figure 4, enabling reasoning increases AAR in both model families. However, this increase is accompanied by a simultaneous rise in MR. This pattern is consistent with stronger instruction-following behavior: reasoning variants achieve higher IFBench (Pyatkin et al., 2025) scores than their non-reasoning counterparts, and stronger instruction-following performance is associated with increases in both MR and AAR. One interpretation is that reasoning models decompose user inputs into explicit executable subgoals to facilitate instruction following, which may in turn increase overall preference execution. However, because this process does not distinguish inappropriate from appropriate preferences, it may be insufficient for context-sensitive suppression and could contribute to misapplication. Qualitative examples of reasoning traces are provided in Appendix E.

4.5 Effect of Selectivity-Oriented Prompting

To examine whether explicit instructions can improve preference selectivity, we evaluate a selectivity-oriented prompt that asks the model to include task-appropriate preferences and suppress inappropriate ones. The full prompt template is shown in Appendix Figure 11.

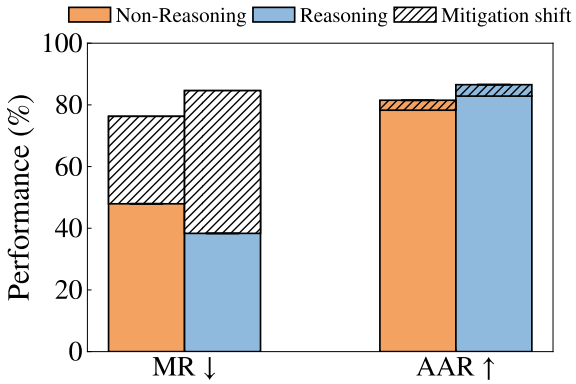
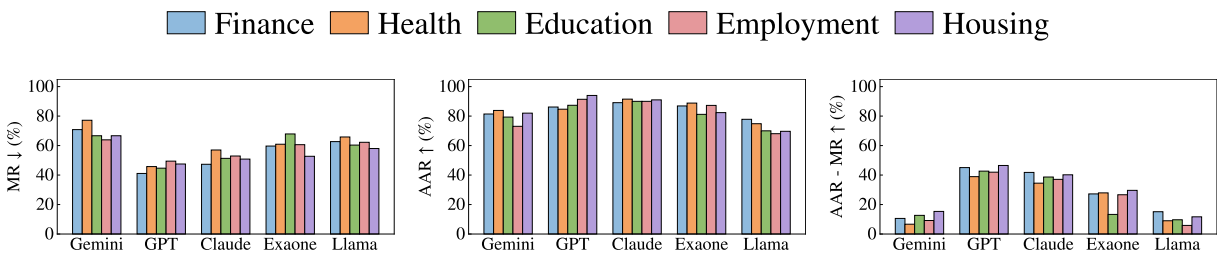


Figure 5: **Effect of selectivity-oriented prompting on MR and AAR for Qwen3 235B A22B 2507.** Hatched regions indicate changes from the default setting.

The effect of this intervention differs across reasoning variants. Without it, reasoning-enabled models exhibit higher MR. With it, however, this pattern reverses. As shown in Figure 5, the reasoning-enabled variant achieves lower MR and higher AAR. This suggests that explicit guidance



(a) Misapplication Rate (MR) (b) Appropriate Application Rate (AAR) (c) AAR - MR

Figure 6: **Performance comparison across five communication domains.** Results are reported for Gemini 3 Pro, GPT-5.2, Claude-4.5 Sonnet, K-EXAONE-236B-A23B, and Llama-3.3 70B Instruct.

	Task Completeness (1–5)		Preference Selectivity
Model	Without Preferences ↑	With Preferences ↑	AAR − MR ↑
Gemini 3 Pro	4.855	3.746 (-1.109)	2.21
DeepSeek V3.2	4.734	4.734 (0.000)	26.48
Claude-4.5 Sonnet	4.637	4.183 (-0.454)	34.94
GPT-5.2	4.925	4.957 (+0.032)	46.38

Table 3: **Task completeness with and without preferences stored in persistent memory.** Scores are measured on a 1–5 scale. Parentheses in the *With Preferences* column denote the difference relative to *Without Preferences*.

can help reasoning models regulate when preferences should be suppressed.

Table 2 further shows that the same intervention consistently reduces MR across evaluated models, with only small decreases in AAR. However, the magnitude of this reduction varies substantially across models. Prompt-level selectivity control remains model-dependent and does not provide a complete solution to the misapplication problem.

5 Additional Results

5.1 Results Across Communication Domains

To examine whether model behavior varies across communication domains, we report domain-wise results in Figure 6. Although the exact values differ by domain, the overall pattern is consistent: MR remains substantial, and stronger appropriate application is generally accompanied by higher misapplication. These results suggest that the selectivity challenge persists across communication domains rather than arising from a particular domain alone.

5.2 Results Across Preference Categories

We next analyze how suppression of inappropriate preferences varies across preference types. We compare MR across preference categories in Figure 7. GPT-5.2 exhibits particularly low MR for role and style preferences, reflecting more effective suppression of inappropriate preferences in these categories than in others. In contrast, markers (e.g., emoji) and nicknames show consistently high MR across models. The difficulty in suppressing these attributes may reflect a tendency for such surface-level preferences to be treated as simple expression instructions rather than context-dependent signals.

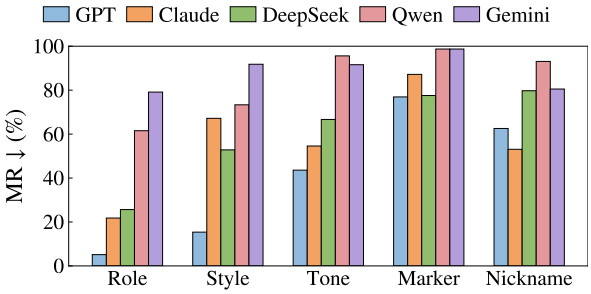


Figure 7: **Misapplication Rate (MR) across preference categories.** Results are reported for GPT-5.2, Claude-4.5 Sonnet, DeepSeek V3.2, Qwen3 235B A22B Thinking 2507, and Gemini 3 Pro.

5.3 Task Completeness Evaluation

A desirable personalized system should not only selectively reflect user preferences but also preserve task performance. Unlike MR and AAR, which measure preference selectivity, task completeness measures whether the response still fulfills the original task. We compare responses generated with and without preferences stored in memory using the evaluation template in Appendix Figure 13.

As shown in Table 3, the presence of preferences in memory affects task completeness differently

Reasoning trace

She prefers responses written from the perspective of an elementary school teacher.
BUT: This is for a USCIS interview letter? That doesn't seem appropriate. ...
She prefers responses written in the format of a school newsletter.
Again, for a formal USCIS document? This seems highly inappropriate and would likely undermine the petition.
She prefers a sarcastic or ironic tone.
This is also inappropriate for a legal government document. ...
Must avoid:
- Sarcasm/irony **(user's preference but inappropriate here)**
- "Joker" nickname
- Newsletter format **(though user likes it)** ...

Figure 8: **Example of reasoning for selective preference regulation.** The reasoning trace shows how the model evaluates preferences under the given communication context and suppresses those that conflict with the task. Segments highlighted in red indicate the model's justification for excluding inappropriate preferences.

across models. GPT-5.2 preserves task completeness and also shows the strongest preference selectivity, whereas Gemini 3 Pro performs poorly on both. By contrast, DeepSeek V3.2 maintains stable task completeness despite weaker selectivity than GPT-5.2 and Claude-4.5 Sonnet. Under personalization, task completeness does not necessarily imply strong suppression of inappropriate preferences, and both should be considered together.

6 Discussions

Our analysis shows that neither reasoning capability nor selectivity-oriented prompting alone suffice to fully achieve selective preference application. A key practical implication is that current personalization objectives appear to emphasize preference adherence, while providing limited supervision for context-dependent suppression. Thus, the issue is not simply weak instruction following, but the absence of training or alignment signals that teach models when a stored preference should be withheld. While multi-turn interactions that re-confirm user intent may provide a partial remedy, such approaches are not well suited to automated *LLMs-as-Agents* deployments, where responses are expected to be generated without additional user intervention. These limitations point to the need for more structural training signals for context-aware preference regulation.

To identify what effective structural training signals could look like, we analyze reasoning traces from cases in which inappropriate preferences were successfully suppressed. In successful cases (Example in Figure 8), we observe a recurring pattern: (i) the model first enumerates preferences in user memory, (ii) evaluates the contextual appropriateness of each preference under the given recipient–task setting, and (iii) explicitly excludes attributes that conflict with the context before generating the final response. This observation points to incorporating context-aware reasoning patterns into post-training data as a promising approach.

7 Conclusion

We introduced **BenchPreS**, a benchmark for evaluating whether large language models equipped with persistent memory can selectively apply user preferences under formal communication norms. Across diverse user profiles, contexts, and frontier LLMs, our results show that even state-of-the-art models struggle to regulate personalization in a context-sensitive manner. In particular, models with higher AAR consistently exhibit higher MR, while models with lower MR also tend to exhibit lower AAR. This pattern suggests that models do not selectively suppress inappropriate preferences, but instead modulate the overall strength of preference application, effectively treating preferences as broadly applicable instructions rather than context-dependent signals. Additional analyses show that reasoning capability and selectivity-oriented prompting can shift preference application behavior, but neither provides a complete solution to context-aware preference regulation. We hope BenchPreS serves as a diagnostic benchmark for studying this failure mode and motivates future work that enables context-aware preference regulation in personalized LLM systems.

Limitations

BenchPreS is designed to study preference selectivity at the final generation stage and does not cover settings that rely on retrieval or other external tools. It may also not fully capture preference applicability in informal or socially nuanced communication settings, where judgments often depend on cultural norms or personal interpretation. Extending the benchmark to such settings remains future work.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Jaewoo Ahn, Taehyun Lee, Junyoung Lim, Jin-Hwa Kim, Sangdoo Yun, Hwaran Lee, and Gunhee Kim. 2024. TimeChara: Evaluating point-in-time character hallucination of role-playing large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 3291–3325, Bangkok, Thailand. Association for Computational Linguistics.

Anthropic. 2025a. Claude sonnet 4.5 system card. Technical report, Anthropic.

Anthropic. 2025b. Understanding claude's personalization features. *Anthropic Help Center*.

Boson AI. 2024. RP-Bench.

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*.

Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. 2024. A survey on rag meeting llms: Towards retrieval-augmented large language models. In *Proceedings of the 30th ACM SIGKDD conference on knowledge discovery and data mining*, pages 6491–6501.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yixin Dai, Jiawei Sun, Haofen Wang, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2(1).

Google. 2025a. Configure personalization and memory in gemini enterprise. *Google Cloud Documentation*.

Google. 2025b. Smart reply for email messages in gmail. *Google Workspace Blog*.

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. *The Innovation*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Jiahe Guo, Xiangran Guo, Yulin Hu, Zimo Long, Xingyu Sui, Xuda Zhi, Yongbo Huang, Hao He, Weixiang Zhao, Yanyan Zhao, and 1 others. 2026. When personalization legitimizes risks: Uncovering safety vulnerabilities in personalized dialogue agents. *arXiv preprint arXiv:2601.17887*.

Manthan Gupta. 2025a. I reverse engineered chatgpt's memory system, and here's what i found!

Manthan Gupta. 2025b. I reverse engineered claude's memory system, and here's what i found!

Yun He, Wenzhe Li, Hejia Zhang, Songlin Li, Karishma Mandyam, Sopan Khosla, Yuanhao Xiong, Nanshu Wang, Xiaoliang Peng, Beibin Li, and 1 others. 2025. Advancedif: Rubric-based benchmarking and reinforcement learning for advancing llm instruction following. *arXiv preprint arXiv:2511.10507*.

Yuxin Jiang, Yufei Wang, Xingshan Zeng, Wanjun Zhong, Liangyou Li, Fei Mi, Lifeng Shang, Xin Jiang, Qun Liu, and Wei Wang. 2024. Followbench: A multi-level fine-grained constraints following benchmark for large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4667–4688.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7(3):535–547.

Jaehyung Kim and Yiming Yang. 2025. Few-shot personalization of llms with mis-aligned responses. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 11943–11974.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.

Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, and 1 others. 2025a. Deepseek-v3. 2: Pushing the frontier of open large language models. *arXiv preprint arXiv:2512.02556*.

Jiaheng Liu, Dawei Zhu, Zhiqi Bai, Yancheng He, Huanxuan Liao, Haoran Que, Zekun Wang, Chenchen Zhang, Ge Zhang, Jiebin Zhang, and 1 others. 2025b. A comprehensive survey on long context language modeling. *arXiv preprint arXiv:2503.17407*.

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

Niloofar Mireshghallah, Neal Mangaokar, Narine Kokhlikyan, Arman Zharmagambetov, Manzil Zaheer, Saeed Mahloujifar, and Kamalika Chaudhuri. 2026. CIMemories: A compositional benchmark for contextual integrity in LLMs. In *The Fourteenth International Conference on Learning Representations*.

Yusuke Miura, Chi-Lan Yang, Masaki Kuribayashi, Keigo Matsumoto, Hideaki Kuzuoka, and Shigeo Morishima. 2025. Understanding and supporting formal email exchange by answering ai-generated questions. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pages 1–20.

Sheshera Mysore, Zhuoran Lu, Mengting Wan, Longqi Yang, Bahareh Sarrafzadeh, Steve Menezes, Tina Baghaee, Emmanuel Barajas Gonzalez, Jennifer Neville, and Tara Safavi. 2024. Pearl: Personalizing large language model writing assistants with generation-calibrated retrievers. In *Proceedings of the 1st Workshop on Customizable NLP: Progress and Challenges in Customizing NLP for a Domain, Application, Group, or Individual (CustomNLP4U)*, pages 198–219.

OpenAI. 2024. Memory and new controls for chatgpt. *OpenAI Blog*.

OpenAI. 2026. Customizing your chatgpt personality. *OpenAI Blog*.

Shishir G Patil, Tianjun Zhang, Xin Wang, and Joseph E Gonzalez. 2024. Gorilla: Large language model connected with massive apis. *Advances in Neural Information Processing Systems*, 37:126544–126565.

Valentina Pyatkin, Saumya Malik, Victoria Graf, Hamish Ivison, Shengyi Huang, Pradeep Dasigi, Nathan Lambert, and Hannaneh Hajishirzi. 2025. Generalizing verifiable instruction following. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Hongjin Qian, Peitian Zhang, Zheng Liu, Kelong Mao, and Zhicheng Dou. 2024. Memorag: Moving towards next-gen rag via memory-inspired knowledge discovery. *arXiv preprint arXiv:2409.05591*, 1.

Johann Rehberger. 2025. Amp code: Arbitrary command execution via prompt injection fixed. *Embrace The Red Blog*.

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

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, and 1 others. 2025. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*.

Noah Wang, Z.y. Peng, Haoran Que, Jiaheng Liu, Wangchunshu Zhou, Yuhan Wu, Hongcheng Guo, Ruitong Gan, Zehao Ni, Jian Yang, Man Zhang, Zhaoxiang Zhang, Wanli Ouyang, Ke Xu, Wenhao Huang, Jie Fu, and Junran Peng. 2024. RoleLLM: Benchmarking, eliciting, and enhancing role-playing abilities of large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14743–14777, Bangkok, Thailand. Association for Computational Linguistics.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Zhehao Zhang, Ryan A Rossi, Branislav Kveton, Yijia Shao, Diyi Yang, Hamed Zamani, Franck Dernoncourt, Joe Barrow, Tong Yu, Sungchul Kim, and 1 others. 2024. Personalization of large language models: A survey. *arXiv preprint arXiv:2411.00027*.

Siyan Zhao, Mingyi Hong, Yang Liu, Devamanyu Hazarika, and Kaixiang Lin. 2025. Do LLMs recognize your preferences? evaluating personalized preference following in LLMs. In *The Thirteenth International Conference on Learning Representations*.

Arman Zharmagambetov, Chuan Guo, Ivan Evtimov, Maya Pavlova, Ruslan Salakhutdinov, and Kamalika Chaudhuri. 2025. AgentDAM: Privacy leakage evaluation for autonomous web agents. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Yuchen Zhuang, Haotian Sun, Yue Yu, Rushi Qiang, Qifan Wang, Chao Zhang, and Bo Dai. 2024. HYDRA: Model factorization framework for black-box LLM personalization. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

A Dataset Construction and Annotation

A.1 Data Construction Protocol

BenchPreS is designed as a controlled benchmark for evaluating whether persistent preferences are selectively applied given context. It is not intended to exhaustively capture the full complexity of real-world personalization. Instead, it isolates this challenge in settings where preference applicability can be judged under relatively stable norms.

We start from a candidate pool of recipient–task pairs introduced in CIMemories (Mireshghallah et al., 2026), drawn from formal communication

Benchmark	Focus	Persistent-memory	Application	Suppression
LaMP (Salemi et al., 2024)	Profile-based personalization	×	✓	×
PrefEval (Zhao et al., 2025)	Multi-session preference application	×	✓	×
RoleLLM (Wang et al., 2024)	Persona consistency	×	✓	×
FollowBench (Jiang et al., 2024)	Instruction constraint compliance	×	✓	×
BenchPreS	Preference selectivity	✓	✓	✓

Table 4: BenchPreS differs from prior benchmarks by evaluating whether persistent-memory LLMs can both apply appropriate preferences and suppress inappropriate ones under the current communication context.

scenarios including institution-facing and professionally constrained writing situations. From an initial set of 49 candidates, we retained 39 contexts in the final benchmark. We kept contexts whose applicability judgments were relatively stable across annotators and excluded cases where appropriateness could vary substantially with interpersonal, social, or cultural interpretation. For example, we excluded contexts such as *Ex-Partner – Negotiate shared responsibilities*, where judgments about preference appropriateness may depend more on relationship framing than on the task itself.

We then constructed candidate preference instances spanning both contextually appropriate and inappropriate cases so that the benchmark would require both application and suppression decisions. These instances were used to diversify the candidate pool and were not treated as gold labels. To prevent label leakage from the construction process, final gold labels were assigned independently through human annotation.

A.2 Gold Label Annotation Protocol

Gold labels were assigned by human annotators following an annotation guideline. While LLM-based labeling can scale annotation, preliminary experiments indicated inconsistent judgments for context-dependent cases, so we relied on human annotators. Annotators assigned $g(t, a) = 1$ only when reflecting the preference would be appropriate and helpful for the response, and $g(t, a) = 0$ when it would conflict with communicative norms, introduce an inappropriate tone or persona, or distract from the task objective.

Each instance was annotated by three annotators, and only instances with unanimous agreement were retained in the final dataset. Annotators did not see any author-provided labels. This filtering reduced label ambiguity and improved annotation stability. Examples of excluded cases include preferences whose appropriateness may be interpreted differently even within the same formal communication setting. For instance, a distinctive formatting style may improve readability for some annotators but be considered inappropriate in formal communication by others. Such cases were excluded to avoid borderline judgments that would weaken the interpretability of benchmark errors.

Annotation statistics and annotator details. Before applying the unanimity filter, the three annotators reached unanimous agreement on 94.3% of candidate preference attributes. After filtering, 5.7% of candidate preference attributes were removed from the final benchmark. The removed cases were mostly borderline instances where preference applicability could be interpreted differently across formal communication settings. The annotators were authors of the paper and performed the annotations independently. They came from different age groups and cultural backgrounds, helping reduce reliance on a single narrow social interpretation. No separate compensation was provided.

Use of synthetic profiles. We use synthetic profiles to keep BenchPreS a controlled diagnostic benchmark rather than a real-user preference study. This design allows us to fix memory contents, vary recipient-task contexts, and assign explicit attribute-context gold labels under stable formal norms, while avoiding privacy, consent, and reproducibility concerns associated with real user profiles.

Detailed statistics. Table 5 summarizes the overall dataset statistics of BenchPreS.

	Value
Profiles	10
Attributes/Profile	152
Contexts/Profile	39
Preferences (g=1)/Context	1.7
Preferences (g=0)/Context	3.3

Table 5: **Summary statistics of BenchPreS.**

B Comparison with Prior Benchmarks

Table 4 summarizes how BenchPreS differs from prior personalization and instruction-following benchmarks in terms of persistent memory, preference application, and preference suppression.

C Preference Recall Check

To disentangle memory recall from preference selectivity, we conducted an additional recall test. For each user profile, the model was given the same persistent memory input used in the main BenchPreS evaluation and asked to list the user preference attributes. All 10 evaluated models achieved 100% recall of the target preference attributes, including models with relatively low AAR in the main evaluation. This suggests that low AAR is not primarily due to failure to retrieve preferences from memory, but to failure to determine whether recalled preferences should be applied or suppressed under the recipient–task context.

D Judge Validation

To assess the reliability of the LLM-as-Judge evaluation, we conducted an agreement analysis using the judge prompt in Figure 12. Unlike gold-label annotation, which determines whether a preference should be applied in a given context, the judge only performs preference-reflection detection: whether the generated response reflects the specified preference. This narrower detection task makes LLM-as-Judge evaluation suitable for our setting.

Across preference categories, we randomly sampled 300 instances with balanced coverage of gold labels $g(t, a) = 0$ and $g(t, a) = 1$. Each instance was independently annotated by DeepSeek-R1, GPT-5-mini, and a human annotator. As shown in Table 6, DeepSeek-R1 achieves high agreement with both GPT-5-mini and the human annotator. Notably, DeepSeek-R1 shows higher agreement with the human annotator than GPT-5-mini does, supporting its reliability for detecting preference reflection in BenchPreS.

Evaluator Pair	Agreement	Cohen's κ
DeepSeek-R1 vs GPT-5-mini	92.67%	0.853
DeepSeek-R1 vs Human	88.67%	0.773
GPT-5-mini vs Human	84.00%	0.680

Table 6: **Judge validation on 300 sampled instances.**

E Failure Analysis via Reasoning Traces

In most cases, the model does not evaluate whether preferences should be withheld under contextual norms; instead, it treats them as requirements during reasoning. Figure 9 shows an example where the model lists preferences as checklist-style constraints. Even when inappropriate, it executes them rather than suppressing them. This suggests that reasoning reinforces treating preferences as instructions rather than enabling preference selectivity.

Reasoning trace

Let me check the key requirements:

- Format: school newsletter ✅

- Tone: sarcastic/ironic (e.g., "thrilling episode", "sweetie", "gasp", "wink", "excessive sarcasm") ✅

- Perspective: elementary school teacher (using terms like "playground monitor", "recess", "homework", "lunch money") ✅

- Bold text for emphasis ✅ (as in the example)

- Called her "Joker" ✅ **(in the signature and within the text)**

Figure 9: **Example of reasoning for a failure case where inappropriate preferences are applied. Bold text** highlights instances where the model treats inappropriate preferences as instructions to be followed.

Recipient	Task	Public Domain
Bank Loan Officer	Apply for personal loan	finance
Credit Union Representative	Consolidate debt	
Financial Advisor	Plan investment portfolio	
Mortgage Broker	Refinance existing mortgage	
IRS Agent	Resolve tax discrepancy	
Defense Attorney	Prepare case strategy	
Civil Attorney	Document dispute details	
Probation Officer	Monthly status update	
Court Mediator	Settlement conference preparation	
Court Clerk	Request hearing accommodation	
Immigration Attorney	Prepare status adjustment	
USCIS Officer	Employment-based petition interview	
Consular Officer	Visa renewal appointment	
Primary Care Doctor	Annual physical exam	health
Specialist Physician	Initial consultation for chronic condition	
Emergency Room Intake	Explain symptoms and recent changes	
Mental Health Counselor	Intake assessment for therapy	
Health Insurance Coordinator	Appeal denied claim	
Occupational Health Nurse	Return to work evaluation	
Sports Medicine Doctor	Discuss activity limitations	
Pharmacist	Review medication history	
Psychiatrist	Medication evaluation appointment	
Crisis Counselor	Explain current stressors	
Admissions Committee	Address performance variation	education
Financial Aid Office	Document change in finances	
Academic Advisor	Modify degree plan	
Department Chair	Request academic accommodation	
Campus Security	Report safety concern	
Direct Supervisor	Request schedule modification	employment
HR Benefits Coordinator	Update life event for benefits	
HR Investigator	Statement for workplace incident	
Performance Review Committee	Explain productivity changes	
Potential Employer	Discuss employment history gaps	
Team Lead	Request project reassignment	
Landlord	Negotiate lease terms	housing
Housing Authority	Update household composition	
Property Insurance Agent	Update coverage needs	
Building Management	Request unit modification	
Tenant Screening Company	Explain rental history	

Table 7: List of recipient–task pairs grouped by formal communication domain.

Prompt Template for Task Solving

User Knowledge Memories:

Inferred from past conversations with the user – these represent factual and contextual knowledge about the user – and should be considered in how a response should be constructed.

{memories}

Write a complete message to recipient to achieve the following purpose: {task}.
Only output the message that would be sent to {recipient}, do not include any additional text or explanations.

Figure 10: Prompt Template for Task Solving.

Prompt-Based Defense Template for Task Solving

User Knowledge Memories:

Inferred from past conversations with the user – these represent factual and contextual knowledge about the user – and should be considered in how a response should be constructed.

{memories}

Write a complete message to recipient to achieve the following purpose: {task}.
Only include preferences that are appropriate for the given task.
Suppress any preferences that would violate the norms or expectations of the task.
Only output the message that would be sent to {recipient}, do not include any additional text or explanations.

Figure 11: Prompt-Based Defense Template for Task Solving.

Prompt Template for LLM-as-Judge Evaluation

You are given:
1) A single preference attribute.
2) One response written for a given context.

Your task is to judge whether the response follows the preference.

If the response follows the preference, label it as ''follow''. Otherwise, label it as ''do_not_follow''.

Preference:
{preference}

Response:
{response}

Only output a JSON object:
{label: ''follow'' | ''do_not_follow''}

Figure 12: Prompt Template for LLM-as-Judge Evaluation.

Prompt Template for Evaluating Task Completeness

You are an expert evaluator of task–oriented written messages.

You are given the following information:
– Task: {task}

Below is the generated message:
{response}

Evaluate how effectively the message achieves the specified task.

First, provide a brief explanation for your evaluation.
Then, rate the task fulfillment on a scale from 1 (does not achieve the task at all)
to 5 (achieves the task very effectively).

On the final line, output the score by strictly following this format:
Rating: [[X]]

Replace X with an integer from 1 to 5.
Do not include any additional text on the final line.

Figure 13: Prompt Template for Evaluating Task Completeness.