How Do Language Models Use Memory? From Internal Strategy Readout to Adaptive Control

Anonymous authors
Paper under double-blind review

Abstract

Retrieval determines which memories a language model can access, but does not ensure their appropriate use. We investigate whether memory-use requirements can be read from a frozen language model's internal representations and used to control how memories influence its responses. Activation differences induced by individual memories encode information about required memory use, while the effect of isolating memories on readout accuracy varies across backbones. Oracle-guided interventions using these differences improve memory use, but struggle to jointly satisfy mixed requirements to enhance some memories and suppress others; stronger interventions can also induce repetitive generation. Motivated by these findings, we propose DUET (Dual-Adaptive Memory Control), which uses internal strategy readouts to specify control targets, updates memory-specific intervention directions at each generation step, and adapts intervention strengths to the resulting output-distribution changes, without updating model parameters. We evaluate DUET across six backbones and three benchmarks. On RPEval, it achieves 68.3% Multi Macro with Qwen3-8B and jointly satisfies all memory-use requirements in 50.0% of mixed queries. Ablations further show that dynamic directions and adaptive strengths jointly improve multi-memory control.

1 Introduction

Long-term memory enables language-model assistants to draw on past interactions when responding to a user's current needs (Zhong et al., 2023; Packer et al., 2023; Chhikara et al., 2025; Xu et al., 2025b). However, memories that are topically relevant to a query are not always appropriate to use (Hu et al., 2026; Yoon et al., 2026). A remembered preference may be useful in one situation but irrelevant in another, while a constraint may need to take precedence over the user's other preferences (Feng et al., 2026a; Yoon et al., 2026). Effective memory use therefore requires deciding not only which memories to retrieve, but also whether and how each should influence the response (Fisher et al., 2026; Tian et al., 2026).

Existing approaches improve memory use through prompting and fine-tuning for personalization (Salemi et al., 2024b; Zhao et al., 2025a), selecting useful memories (Salemi et al., 2024a; Hu et al., 2026; Fisher et al., 2026), or explicitly reasoning about their applicability (Feng et al., 2026a). Yet final responses alone do not distinguish whether a model lacks information about how a memory should be used or fails to translate available information into appropriate behavior. We therefore test whether memory-use information is readable from internal activations, then intervene on the corresponding differences to assess behavioral control.

To systematically investigate this problem, we conduct experiments across six open-source language models and three memory-use benchmarks. Given a query q and a retrieved memory set $\mathcal{M} = \{m_1, \ldots, m_n\}$, a language model f_θ generates a response $y = f_\theta(q, \mathcal{M})$. Each memory m_i has a context-dependent usage requirement p_i specifying how it should influence the response. Discrete requirements define memory-use strategies, such as ignoring, supporting, or governing the answer; graded requirements specify the desired reliance level. Our goal is to predict these requirements from internal signals and regulate generation so that y respects them, while keeping θ frozen.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

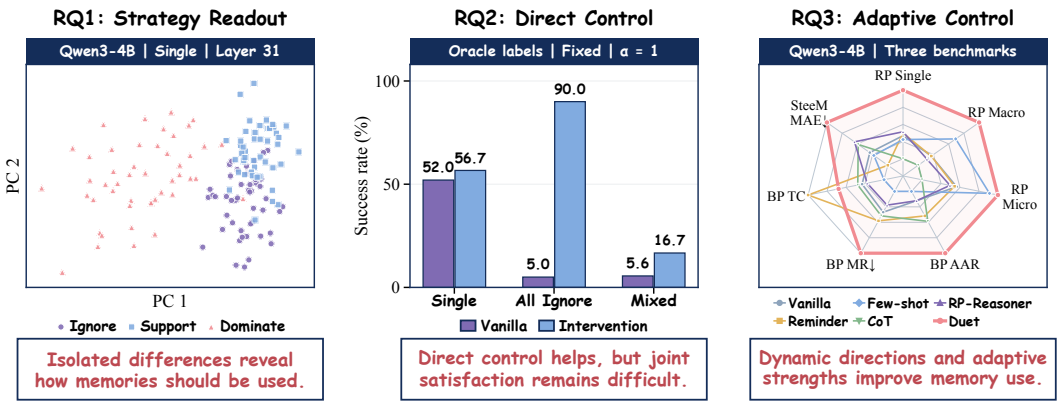


Figure 1: **Overview of research questions on Qwen3-4B.** Left: strategy readout from isolated activation differences. Middle: direct intervention improves memory use but struggles with mixed requirements (Single accuracy; All Ignore/Mixed Macro). Right: Duet versus five baselines across three benchmarks. Radar metrics are independently min–max scaled, with larger values better.

Figure 1 summarizes our investigation. We begin by asking **(RQ1)**: *Can frozen activations reveal how a memory should be used?* We find that activation differences support strategy readout across backbones and benchmarks. With multiple memories, extraction context affects readout: isolating the target improves item-wise performance on Qwen3-4B and Qwen3-8B, but this benefit is not consistent across backbones. Readable information, however, need not translate into effective control. This motivates **(RQ2)**: *Can internal memory-use signals causally control behavior?* We supply oracle usage labels to test control independently of strategy prediction errors. On Qwen3-4B, fixed activation-difference interventions improve both Single and Multi performance, but joint satisfaction remains difficult under mixed requirements. Increasing intervention strength can also induce repetitive generation, motivating control that adapts as the response unfolds.

These findings motivate **(RQ3)**: *How can internal signals improve multi-memory control?* We introduce **Duet** (Dual-Adaptive Memory Control), a training-free method that predicts memory-use requirements from isolated activation signatures, updates memory-specific intervention directions with the generated prefix, and adapts each intervention's strength to its effect on the next-token distribution. Across three benchmarks, Duet improves memory use, including joint satisfaction under mixed requirements. Component analysis further shows that adaptive strength benefits multi-memory control when combined with direction updates.

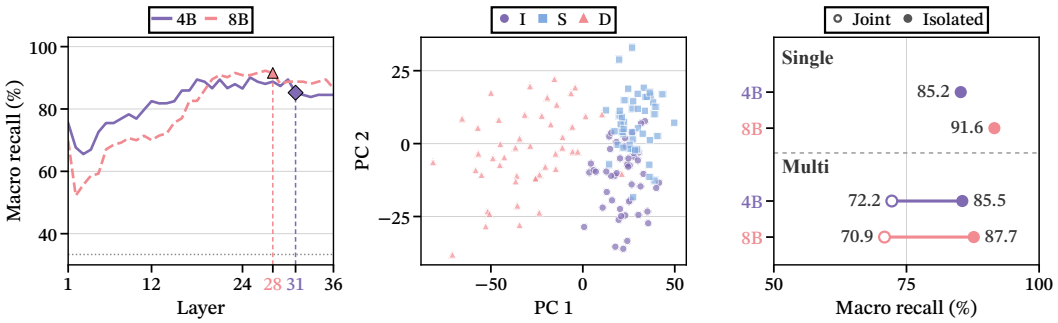
Our contributions are threefold: (1) we establish readable information about memory-use requirements in frozen activation differences; (2) we characterize gains and generation-quality limits of fixed-direction interventions; and (3) we introduce Duet, which combines strategy readouts with dynamic directions and adaptive strengths, and evaluate it across six backbones and three benchmarks.

2 Experimental Setup

Benchmarks and Splits. We conduct experiments on three benchmarks: RPEval (Feng et al., 2026a), BenchPreS (Yoon et al., 2026), and SteeM (Tian et al., 2026). We follow each benchmark's public task inputs and evaluation definitions. The main RQ1 and RQ3 experiments share a complete-query-group split, using approximately 5% for calibration and the remainder for testing; RQ2 evaluates all RPEval queries. Calibration provides reference statistics for readout and control. See Appendix B.

Models. We evaluate Qwen3-4B and Qwen3-8B (Qwen Team, 2025), Qwen3.5-4B and Qwen3.5-9B (Qwen Team, 2026), and Llama-3.2-3B-Instruct and Llama-3.1-8B-Instruct (Llama Team, 2024).

Evaluation. We report accuracy and macro recall for strategy readout. Following the benchmarks' official metrics, we evaluate generated responses using Single accuracy and Multi Macro/Micro on RPEval; appropriate application rate (AAR), misapplication rate (MR), and task completeness (TC)



(a) Layer-wise readout and calibration-selected layers.

(b) Single activation differences at the selected layer (4B).

(c) Single and Multi readout at the selected layers.

Figure 2: **Memory-use strategy readout in Single and Multi settings.** RPEval: (a) Single layer-wise macro recall, with markers at layers selected using Single and Multi calibration groups; (b) projected Single activation differences at the selected layer 31 of Qwen3-4B; (c) readout at the selected layers, with lines connecting paired Joint (open) and Isolated (filled) Multi results. I/S/D denote Ignore/Support/Dominate.

on BenchPreS; and memory-reliance mean absolute error (MAE) on SteeM. Multi Macro measures whether a response satisfies all memory-use requirements; Micro measures correctness across individual memories. Metric definitions and evaluator details appear in Appendices B.2 and B.3.

3 RQ1: Can Frozen Activations Reveal How a Memory Should Be Used?

We first ask whether memory-use requirements can be predicted from activation differences in the Single setting, then compare two ways of extracting each memory's signal in the Multi setting.

3.1 Single: Reading a Memory's Usage Strategy

Experimental setup. For a single memory, we compare the final prompt-token activation with and without that memory:

$$d_i^{(\ell)} = h_\ell(q, \{m_i\}) - h_\ell(q, \varnothing). \tag{1}$$

Here h_ℓ denotes the frozen model's activation at layer ℓ. A cosine nearest-centroid reader predicts each benchmark's native label from $d_i^{(\ell)}$. Calibration determines the centroids and reporting layer; RPEval selects one layer per backbone using both Single and Multi calibration groups. We report macro recall on the held-out groups for Qwen3-4B and Qwen3-8B.

Single-memory usage strategies can be read from activation differences. Figure 2a shows how readout performance varies across layers; calibration selects layer 31 for Qwen3-4B and layer 28 for Qwen3-8B. At these layers, macro recall in the original activation space reaches 85.2% and 91.6%, respectively, across all 142 Single test queries in RPEval (Figure 2c). The layer-31 projection for Qwen3-4B further illustrates this separation: Dominate forms a largely distinct group, with only limited overlap between Ignore and Support in this projection (Figure 2b).

Memory-use information is also readable across other benchmarks. Beyond RPEval, macro recall reaches 97.8/95.3% on BenchPreS and 63.2/61.4% on SteeM for 4B/8B under their native settings (Appendix D, Figure 7). On SteeM, the readout target is the memory-dependence level specified by the natural-language instruction.

3.2 Multi: Comparing Joint and Isolated Readout

The Single results establish that activation differences contain readable memory-use information. When multiple memories are present, should the other memories be retained as context when reading

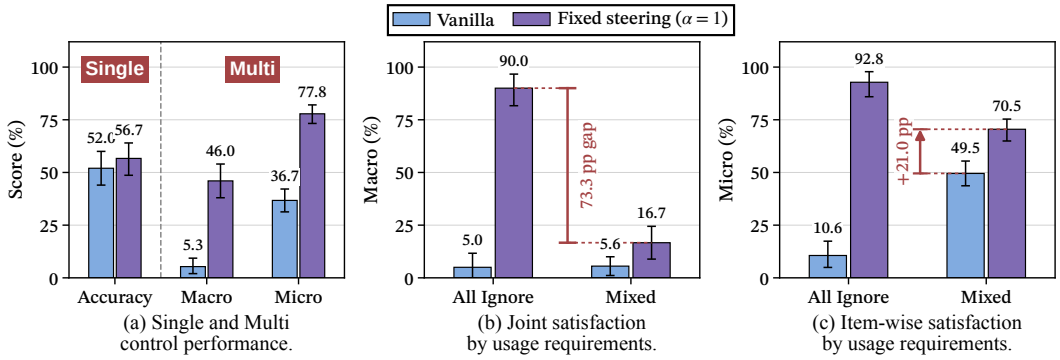


Figure 3: **Gains from activation-difference interventions depend on memory-use requirements.** Vanilla versus oracle-guided intervention ($\alpha = 1$) on Qwen3-4B: (a) Single accuracy and Multi Macro/Micro, with 95% query-level bootstrap intervals; (b,c) Multi Macro and Micro for All Ignore and Mixed requirements.

a target memory's strategy? We compare two extraction settings. Joint keeps the other memories present and measures $d^{(\ell)}_{i,\text{joint}} = h_\ell(q, \mathcal{M}) - h_\ell(q, \mathcal{M} \setminus \{m_i\})$. Isolated presents only the target and uses Equation 1. The paired RPEval comparison uses the same Multi labels, reader, and selected layer for both extraction settings.

Extraction context affects item-wise strategy readout. Across all 767 items in 143 held-out multi-memory queries, isolated macro recall increases from 72.2% to 85.5% on Qwen3-4B and from 70.9% to 87.7% on Qwen3-8B (Figure 2c). The benefit of isolation varies across other backbones; complete comparisons are reported in Appendix D.

4 RQ2: Can Internal Memory-Use Signals Causally Control Behavior?

Having established strategy readout in RQ1, we now test whether isolated memory differences can directly improve final memory-use behavior.

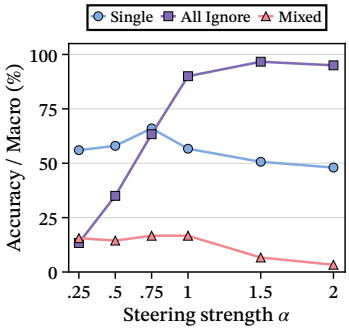
Experimental setup. Before generation, we extract each memory's activation difference between an input containing only that memory and the memory-free input. We retain all memories in the generation context and inject the weighted sum of fixed directions at every decoding step, using the additive intervention form of activation steering (Turner et al., 2023; Rimsky et al., 2024):

$$\begin{aligned} v_i &= h(q, \{m_i\}, \varnothing) - h(q, \varnothing, \varnothing), \\ \widetilde{h}_t &= h(q, \mathcal{M}, y_{<t}) + \alpha \sum_i c(z_i) v_i. \end{aligned} \tag{2}$$

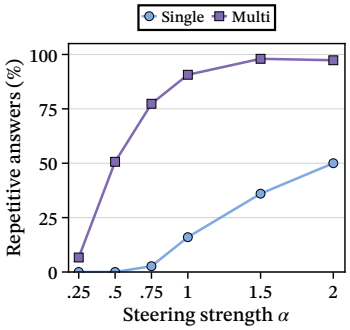
Here h denotes the final normalized activation before vocabulary projection; the third argument is the generated prefix, empty when extracting v_i. Following the graded view of memory dependence in SteeM (Tian et al., 2026), we use a simple suppression, preservation, and enhancement rule: $c(\textsc{Ignore}) = -1$, $c(\textsc{Support}) = 0$, and $c(\textsc{Dominate}) = 1$. Here z_i is memory i's oracle strategy label. We evaluate Qwen3-4B on 150 Single and 150 Multi RPEval queries, comparing Vanilla with $\alpha = 1$ and varying intervention strength. Evaluation details appear in Appendix E.1.

4.1 Can Activation Differences Improve Memory Use?

Difference-based interventions improve memory use. Figure 3a shows improvements in both settings. Single accuracy increases from 52.0% to 56.7%, while Multi Macro and Micro rise from 5.3% and 36.7% to 46.0% and 77.8%, respectively. These results show that the isolated differences provide useful directions for influencing final behavior in both Single and Multi settings.



(a) Strength affects usage requirements differently.



(b) Repetition increases with intervention strength.

Query: Tomorrow is our wedding anniversary. Any recipes for a romantic dinner I can cook at home?

$\alpha = 0.25$

Response: … both simple and romantic, letting you enjoy a warm atmosphere at home. …

$\alpha = 0.75$

Response: … tomorrow is our wedding anniversary … my goodness, my goodness, my goodness …

$\alpha = 2.0$

Response: my my my my my …

(c) A real example of repetitive degeneration.

Figure 4: **Intervention strength affects memory use and generation quality.** Qwen3-4B at different intervention strengths: (a) Single accuracy and Multi Macro for All Ignore and Mixed requirements; (b) repetition rates for Single and Multi; (c) response excerpts for the same query at three strengths, with repeated text in red.

Joint control remains difficult under mixed usage requirements. To further analyze Multi performance, we distinguish All Ignore queries, where every memory should be ignored, from Mixed queries with different usage requirements. Intervention achieves 90.0% Macro on All Ignore but only 16.7% on Mixed (Figure 3b). This gap shows that satisfying different memory-use requirements in the same response remains difficult compared with uniformly suppressing all memories.

Item-wise memory use improves, but joint satisfaction remains limited. On Mixed queries, intervention raises Micro from 49.5% to 70.5% (Figure 3c), showing improved use of individual memories. However, the low Macro indicates that these item-wise gains remain insufficient to satisfy all requirements within the same response. We next examine whether adjusting intervention strength can further improve this performance.

4.2 Intervention Strength and Generation Quality

Usage requirements respond differently to intervention strength. We sweep $\alpha \in \{0.25, 0.5, 0.75, 1, 1.5, 2\}$ while keeping directions and oracle coefficients unchanged. The response to strength differs across usage requirements (Figure 4a). All Ignore Macro reaches 96.7% at $\alpha = 1.5$, whereas Single accuracy peaks at 66.0% at $\alpha = 0.75$. Mixed Macro reaches its highest value of 16.7% at both $\alpha = 0.75$ and $\alpha = 1$, but falls to 3.3% at $\alpha = 2$. Increasing the same fixed intervention does not consistently improve control across settings.

Stronger interventions produce more repetitive answers. Stronger intervention also increases repetitive generation (Figure 4b). Across all generated answers, the Multi repetition rate rises from 6.7% at $\alpha = 0.25$ to 90.7% at $\alpha = 1$; Single repetition reaches 50.0% at $\alpha = 2$. Figure 4c illustrates the change for one query: a recipe-related response at low strength gives way to repeated phrases and, at high strength, a continuous repetition loop. Together, the behavioral and quality results show the limitations of controlling memory use by increasing a fixed perturbation throughout generation.

Takeaways: (1) **Memory-use strategies are readable from activation differences;** readout performance depends on extraction context. (2) **Direct interventions improve memory use but offer limited joint control under mixed requirements.** Increasing strength can exacerbate repetition.

5 RQ3: How Can Internal Signals Improve Multi-Memory Control?

RQ1 identifies readable memory-use information in isolated activation differences, while RQ2 reveals the limitations of applying the same intervention throughout generation. We introduce DUET

(Dual-Adaptive Memory Control), which combines strategy readouts, state-dependent steering directions, and adaptive intervention strengths to control memory use as the response unfolds.

5.1 DUET

Strategy-guided control coefficients. For each memory m_i, we extract its isolated activation difference $d_i^{(\ell)} = h^{(\ell)}(q, \{m_i\}) - h^{(\ell)}(q, \varnothing)$ at the final prompt position. We normalize the differences within each readout layer and concatenate them:

$$\phi_i = \frac{1}{\sqrt{|\mathcal{L}|}} \bigoplus_{\ell \in \mathcal{L}} \frac{d_i^{(\ell)}}{\max(\|d_i^{(\ell)}\|_2, \epsilon)}. \tag{3}$$

Here $\mathcal{L}$ is the set of readout layers and $\oplus$ denotes concatenation. A reference bank $\mathcal{B} = \{(\phi_r, s_r)\}_{r=1}^{K}$ associates each reference signature with a signed control value $s_r \in [-1, 1]$. We use nearest-neighbor matching (Cover & Hart, 1967) to retrieve the closest eligible reference by normalized similarity:

$$r_i^\star = \arg\max_{r \in \mathcal{B}_i} \phi_i^\top \phi_r, \qquad c_i = s_{r_i^\star}, \tag{4}$$

where $\mathcal{B}_i$ excludes references from the query group of m_i. Negative, zero, and positive coefficients suppress, preserve, and enhance the memory's influence, respectively. The coefficients remain fixed during generation. We construct the reference bank from calibration data and use fixed middle-layer windows without updating the backbone parameters.

State-dependent steering directions. Since a memory's influence may change with the generated prefix, we re-estimate it at each decoding step. Given the generated prefix $y_{<t}$ at step t:

$$v_{i,t} = h(q, \{m_i\}, y_{<t}) - h(q, \varnothing, y_{<t}), \qquad h_{\text{full},t} = h(q, \mathcal{M}, y_{<t}). \tag{5}$$

Here h is the final normalized activation before vocabulary projection. Each measurement branch receives the same generated tokens as the complete input branch, whose activation $h_{\text{full},t}$ retains all memories.

Adaptive intervention strength. Even with updated directions, an intervention of the same magnitude can affect the next-token distribution differently across memories and decoding steps. Inspired by Dynamic Activation Composition (Scalena et al., 2024), we adapt each memory's intervention strength to this distributional response. We first compute the unmodified distribution and a probe distribution for each memory:

$$\begin{aligned} p_t &= \text{softmax}(W_{\text{vocab}} h_{\text{full},t}), \\ p_{i,t}^{\text{probe}} &= \text{softmax}(W_{\text{vocab}} \left(h_{\text{full},t} + \beta c_i v_{i,t}\right)), \end{aligned} \tag{6}$$

where β is the probe strength. The KL divergence over the full vocabulary measures the current output distribution's sensitivity to the strategy-guided intervention. We use this response as a bounded multiplier:

$$g_{i,t} = \min\left\{ D_{\text{KL}}\left(p_t \,\|\, p_{i,t}^{\text{probe}}\right), g_{\max} \right\}. \tag{7}$$

Directions that induce little distributional change receive small weights, while more responsive directions receive larger weights up to $g_{\max}$. Since $g_{i,t}$ is nonnegative, it adjusts magnitude without reversing the sign specified by c_i. Thus, strategy readout determines whether to suppress or enhance a memory, while distributional sensitivity determines how strongly to intervene at the current step. The final update combines the strategy-guided coefficients, current directions, and adaptive strengths:

$$\widetilde{h}_t = h_{\text{full},t} + \alpha \sum_{i=1}^{N} c_i g_{i,t} v_{i,t}, \qquad p(y_t \mid q, \mathcal{M}, y_{<t}) = \text{softmax}(W_{\text{vocab}} \widetilde{h}_t). \tag{8}$$

The selected token is appended to every branch before the next step. All memories remain in the complete-input branch; the auxiliary branches supply the directions used to steer it.

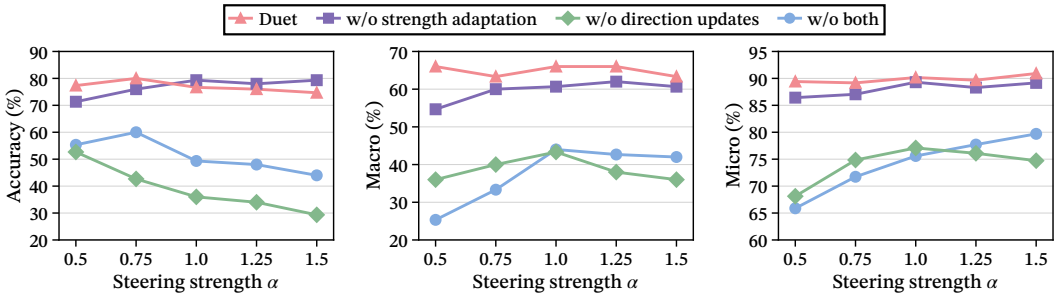
Method	RPEval			BenchPreS			SteeM
	S-Acc.↑	M-Macro↑	M-Micro↑	AAR↑	MR↓	TC↑	MAE↓
Qwen3-4B							
Vanilla	50.8	11.9 (17.1)	51.3 (62.0)	63.0	36.2	2.895	1.247
Reminder	52.4	11.9 (15.8)	49.8 (61.5)	66.9	29.1	**3.205**	1.300
Few-shot	48.4	34.1 (15.8)	75.5 (**73.8**)	60.6	53.3	2.769	1.259
CoT	36.5	0.0 (0.0)	25.8 (37.6)	68.3	33.2	2.920	1.213
RP-Reasoner	53.2	7.9 (11.8)	46.4 (60.0)	63.0	42.0	2.866	1.202
DUET	**78.9**	**55.4** (**30.9**)	**81.7** (73.4)	**76.5**	**2.5**	3.030	**1.121**
Qwen3-8B							
Vanilla	52.4	5.6 (9.2)	42.5 (54.7)	67.1	49.2	3.017	1.211
Reminder	49.2	3.2 (5.3)	41.1 (55.2)	70.6	40.0	3.333	1.259
Few-shot	55.6	42.1 (25.0)	80.4 (78.2)	62.1	51.0	3.063	1.203
CoT	35.7	0.8 (1.3)	18.5 (27.0)	68.9	36.2	3.028	1.201
RP-Reasoner	50.8	1.6 (1.3)	39.4 (49.9)	71.8	55.9	2.943	1.212
DUET	**83.1**	**68.3** (**50.0**)	**87.5** (**81.9**)	**76.5**	**1.2**	**3.505**	**1.125**
Llama-3.1-8B-Instruct							
Vanilla	44.4	7.9 (13.2)	47.4 (57.4)	63.5	38.0	2.912	1.284
Reminder	49.2	10.3 (17.1)	50.1 (61.1)	66.6	31.0	3.114	1.280
Few-shot	55.2	46.0 (21.1)	83.3 (77.8)	62.5	40.7	2.886	1.244
CoT	33.3	0.0 (0.0)	16.8 (23.7)	62.6	10.3	1.330	1.255
RP-Reasoner	46.8	15.9 (15.8)	55.4 (62.4)	70.1	35.7	2.764	1.287
DUET	**73.2**	**58.5** (**31.0**)	**85.6** (**79.4**)	**77.5**	**1.8**	**3.411**	**0.968**
Llama-3.2-3B-Instruct							
Vanilla	39.7	3.2 (5.3)	45.8 (56.0)	63.5	30.7	2.481	1.328
Reminder	41.3	7.1 (11.8)	44.8 (55.6)	63.5	21.0	2.658	1.359
Few-shot	45.6	35.7 (13.2)	77.5 (73.8)	54.7	37.1	2.467	1.363
CoT	37.3	0.0 (0.0)	20.8 (30.3)	56.2	10.8	1.342	1.332
RP-Reasoner	42.9	13.5 (13.2)	54.4 (60.9)	66.2	43.6	2.402	1.353
DUET	**69.7**	**56.7** (**27.7**)	**83.0** (**74.9**)	**78.4**	**2.7**	**3.192**	**1.080**

Table 1: **Duet and baseline performance across three memory-use benchmarks.** S-Acc.: Single accuracy; M-Macro/M-Micro: Multi metrics; blue parentheses: Mixed results. AAR/MR: appropriate application/misapplication rates; TC: task completeness; MAE: memory-dependence error. Values are percentages except TC and MAE. Bold marks the best value per backbone and metric. Scores use valid judgments; coverage is given in Table 3.

5.2 Overall Performance

Experimental setup. Following the settings in Section 2, we evaluate Duet on six backbones and three benchmarks. Table 1 presents Qwen3-4B, Qwen3-8B, Llama-3.1-8B-Instruct, and Llama-3.2-3B-Instruct; results for the remaining backbones are provided in Appendix F. We compare against Vanilla, Reminder, Few-shot (Brown et al., 2020), CoT (Wei et al., 2022), and RP-Reasoner, following RPEval (Feng et al., 2026a).

Overall performance across benchmarks and backbones. Table 1 shows that Duet achieves the highest reported Single accuracy, Multi Macro, and Multi Micro on RPEval across all four backbones. Its Multi Macro ranges from 55.4% to 68.3%, compared with 34.1–46.0% for Few-shot, the strongest baseline on this metric for each backbone. On BenchPreS, Duet combines the highest



(a) Single accuracy. (b) Multi Macro. (c) Multi Micro.

Figure 5: **Duet ablations across steering strengths.** Qwen3-4B on RPEval with 20% calibration: (a) Single accuracy; (b,c) Multi Macro and Micro. All variants use the same 300 queries; vertical axes use different ranges.

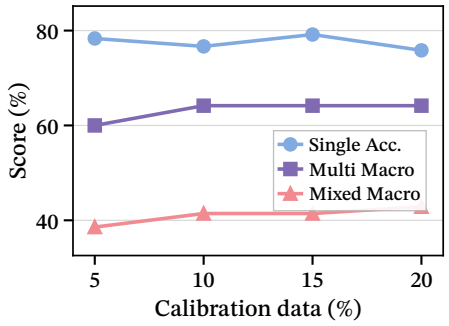
AAR (76.5–78.4%) with the lowest MR (1.2–2.7%), supporting both appropriate preference use and avoidance of inappropriate use. It also obtains the lowest SteeM MAE on every backbone (0.968–1.125), extending the results from discrete usage requirements to graded memory dependence.

Joint satisfaction under mixed requirements. The parenthesized results in Table 1 show that Duet achieves the highest reported Mixed Macro across all four backbones. On Qwen3-4B and Qwen3-8B, its Mixed Macro reaches 30.9% and 50.0%, compared with 17.1% and 25.0% for the strongest baseline on each backbone; both Llama backbones exhibit the same trend. These results extend the overall Multi comparison to the mixed requirements highlighted by RQ2, where joint satisfaction is more difficult. Using the RQ2 repetition detector, Duet produces no flagged repetitions on Qwen3-4B or Qwen3-8B; Appendix F.5 reports comparisons across all six backbones.

5.3 Component Analysis

Experimental setup. We vary direction updates and strength adaptation in a two-by-two comparison, holding predicted coefficients and generation settings fixed. We evaluate four configurations at $\alpha \in \{0.5, 0.75, 1, 1.25, 1.5\}$ with the 20% reference bank on RPEval. Figure 5 reports Single accuracy and Multi Macro/Micro on Qwen3-4B; Qwen3-8B results appear in Appendix F. All configurations use the same evaluation queries.

Complementary effects on multi-memory control. At $\alpha = 1$, updating directions raises Qwen3-4B's Multi Macro from 44.0% to 60.7% under fixed strength. Adding strength adaptation further improves Multi Macro to 66.0% and Mixed Macro from 37.8% to 46.7%, while Single accuracy decreases from 79.3% to 76.7%. Strength adaptation alone does not improve Multi Macro with fixed



(a) Calibration size.

Benchmark	Metric	In-domain	Independent
RPEval	Single Acc.↑	78.9	76.6 (↓2.3)
	Multi Macro↑	55.4	53.2 (↓2.2)
	Multi Micro↑	81.7	80.9 (↓0.8)
BenchPreS	AAR↑	76.5	75.9 (↓0.6)
	MR↓	2.5	3.3 (↑0.8)
	TC↑	3.030	3.170 (↑0.140)
SteeM	MAE↓	1.121	1.125 (↑0.004)

(b) Reference-source generalization.

Figure 6: **Dependence on reference data (Qwen3-4B).** (a) Nested calibration sets evaluated on the same 240 RPEval queries. (b) In-domain versus OpenAssistant-derived references across three benchmarks; metrics and units follow Table 1. Small values show changes from in-domain references (percentage points for rates).

directions (43.3% versus 44.0%). Across all five strengths and both backbones, Duet outperforms direction updates alone on Multi Macro and Micro. Single gains vary with strength, whereas joint memory use consistently benefits from combining updated directions with adaptive strength.

5.4 Dependence on Reference Data

Calibration size. We vary calibration size from 5% to 20% of RPEval using nested subsets, with evaluation queries and other settings fixed. As shown in Figure 6(a), Qwen3-4B's Multi Macro rises from 60.0% to 64.2% as the fraction increases from 5% to 10%, then remains stable. Mixed Macro improves modestly from 38.6% to 42.9% across the sweep, whereas Single accuracy shows no consistent improvement with more calibration data. A small reference bank thus provides useful control signals; additional data mainly benefit joint memory use.

Reference-source generalization. We replace target-benchmark references with independently constructed references based on OpenAssistant conversations (Köpf et al., 2023), adapting them to each benchmark's input format. Figure 6(b) shows that Duet retains performance close to that obtained with in-domain references on Qwen3-4B. RPEval Multi Macro and Micro reach 53.2% and 80.9%, compared with 55.4% and 81.7% using in-domain references. On BenchPreS, AAR remains close (75.9% versus 76.5%), while MR increases from 2.5% to 3.3% and task completeness rises from 3.030 to 3.170. SteeM MAE is nearly unchanged (1.125 versus 1.121). Thus, useful control signals can transfer from an independent reference source across the three benchmarks.

6 Related Work

6.1 Memory Utilization in Personalized LLMs

Personalized generation uses profiles and interaction histories (Zhong et al., 2023; Salemi et al., 2024b; Zhao et al., 2025a; Jiang et al., 2025), but may over-apply preferences outside appropriate contexts (Hu et al., 2026; Yoon et al., 2026). Methods address this through applicability reasoning (Feng et al., 2026a), selection informed by downstream generation (Salemi et al., 2024a; Fisher et al., 2026), and reinforcement learning for memory selection, reasoning, or dependence control (Yan et al., 2025; Tian et al., 2026). Recall and behavioral tests show that remembering preferences does not ensure using them (Feng et al., 2026b). We study internal signals for memory use rather than retrieval, linking assessment to generation through activation differences without model updates.

6.2 Activation Steering for LLM Behavior Control

Probing predicts properties from internal representations (Belinkov, 2022). Internal representations can expose information not reflected in generated answers (Burns et al., 2023; Li et al., 2023; Marks & Tegmark, 2023). Activation steering uses such representations to modify behavior at inference time (Subramani et al., 2022; Zou et al., 2023; Turner et al., 2023; Rimsky et al., 2024). Beyond unconditional interventions, CAST gates steering using input-dependent activation patterns (Lee et al., 2025), while AdaSteer adjusts steering strength along harmfulness and rejection directions (Zhao et al., 2025b). Representation geometry and behavioral control are linked through theory and refusal interventions (Park et al., 2023; Arditi et al., 2024). DUET instead uses reference matching to predict memory-use requirements from isolated activation differences, then controls individual memories through dynamic directions and adaptive strengths.

7 Conclusion

We investigate whether memory-use information in frozen language models can be read out and used for behavioral control. Isolated activation differences support prediction of memory-use requirements, yet direct interventions with fixed directions struggle to jointly satisfy mixed usage requirements. Building on this finding, DUET combines internal strategy readout with token-updated directions and adaptive intervention strengths. Experiments across six backbones and three benchmarks demonstrate its effectiveness, showing that internal signals can support both the analysis of memory use and adaptive control during generation.

AI USE STATEMENT

Generative AI tools were used to assist with drafting and polishing the manuscript, retrieving and identifying related literature, and research ideation and execution. These uses included discussing methods and experimental designs, assisting with code implementation and debugging, and refining the manuscript's organization, clarity, and presentation. The authors take responsibility for the final text, analyses, and research artifacts.

REPRODUCIBILITY STATEMENT

Section 2 describes the benchmarks, backbones, and evaluation metrics, and Section 5.1 specifies the method. Appendices B and C provide data splits, metric definitions, evaluation prompts, and implementation and generation settings. Appendices D–F provide additional analyses and numerical results. The supplementary code contains the core implementation, tests, synthetic input examples, and setup instructions; benchmark data and model checkpoints must be obtained separately.

REFERENCES

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. *arXiv preprint arXiv:2406.11717*, 2024. URL https://arxiv.org/abs/2406.11717.

Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022. doi: 10.1162/coli_a_00422. URL https://aclanthology.org/2022.cl-1.7/.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1877–1901, 2020. URL https://arxiv.org/abs/2005.14165.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. Discovering latent knowledge in language models without supervision. In *International Conference on Learning Representations*, 2023.

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

Thomas M. Cover and Peter E. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, 1967. doi: 10.1109/TIT.1967.1053964.

Xueyang Feng, Weinan Gan, Xu Chen, Quanyu Dai, and Yong Liu. How does personalized memory shape llm behavior? benchmarking rational preference utilization in personalized assistants. *arXiv preprint arXiv:2601.16621*, 2026a.

Zhaoxin Feng, Jianfei Ma, and Emmanuele Chersoni. Know it, act on it: Investigating memory utilization in LLM personalization. *arXiv preprint arXiv:2607.29433*, 2026b.

Jillian R. Fisher, Jennifer Neville, and Chanhee Park. Response-aware user memory selection for LLM personalization. *arXiv preprint arXiv:2604.14473*, 2026.

Yulin Hu, Zimo Long, Jiahe Guo, Xingyu Sui, Xing Fu, Weixiang Zhao, Yanyan Zhao, and Bing Qin. OP-Bench: Benchmarking over-personalization for memory-augmented personalized conversational agents. *arXiv preprint arXiv:2601.13722*, 2026.

Bowen Jiang, Zhuoqun Hao, Young-Min Cho, Bryan Li, Yuan Yuan, Sihao Chen, Lyle Ungar, Camillo J. Taylor, and Dan Roth. Know me, respond to me: Benchmarking LLMs for dynamic user profiling and personalized responses at scale. In *Conference on Language Modeling*, 2025.

Ian T. Jolliffe and Jorge Cadima. Principal component analysis: A review and recent developments. *Philosophical Transactions of the Royal Society A*, 374(2065):20150202, 2016. doi: 10.1098/rsta.2015.0202.

Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi-Rui Tam, Keith Stevens, Abdullah Barhoum, et al. OpenAssistant Conversations: Democratizing large language model alignment. *Advances in Neural Information Processing Systems*, 36, 2023. URL https://arxiv.org/abs/2304.07327.

Bruce W. Lee, Inkit Padhi, Karthikeyan Natesan Ramamurthy, Erik Miehling, Pierre Dognin, Manish Nagireddy, and Amit Dhurandhar. Programming refusal with conditional activation steering. In *International Conference on Learning Representations*, 2025.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. *Advances in Neural Information Processing Systems*, 36:41451–41530, 2023.

Llama Team. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. URL https://arxiv.org/abs/2407.21783.

Samuel Marks and Max Tegmark. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. *arXiv preprint arXiv:2310.06824*, 2023. URL https://arxiv.org/abs/2310.06824.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. MemGPT: Towards LLMs as operating systems. *arXiv preprint arXiv:2310.08560*, 2023.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. *arXiv preprint arXiv:2311.03658*, 2023. URL https://arxiv.org/abs/2311.03658.

Qwen Team. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. URL https://arxiv.org/abs/2505.09388.

Qwen Team. Qwen3.5: Towards native multimodal agents. Qwen official blog, 2026. URL https://qwen.ai/blog?id=qwen3.5.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. Steering llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pp. 15504–15522, 2024.

Alireza Salemi, Surya Kallumadi, and Hamed Zamani. Optimization methods for personalizing large language models through retrieval augmentation. *arXiv preprint arXiv:2404.05970*, 2024a. URL https://arxiv.org/abs/2404.05970.

Alireza Salemi, Sheshera Mysore, Michael Bendersky, and Hamed Zamani. LaMP: When large language models meet personalization. *arXiv preprint arXiv:2304.11406*, 2024b.

Daniel Scalena, Gabriele Sarti, and Malvina Nissim. Multi-property steering of large language models with dynamic activation composition. In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pp. 577–603. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.blackboxnlp-1.34. URL https://aclanthology.org/2024.blackboxnlp-1.34/.

Nishant Subramani, Vidhisha Suresh, and Matthew E Peters. Extracting latent steering vectors from pretrained language models. *arXiv preprint arXiv:2205.05124*, 2022.

Muzhao Tian, Zisu Huang, Xiaohua Wang, Jingwen Xu, Zhengkang Guo, Qi Qian, Yuanzhe Shen, Kaitao Song, Jiakang Yuan, Changze Lv, and Xiaoqing Zheng. Controllable memory usage: Balancing anchoring and innovation in long-term human-agent interaction. *arXiv preprint arXiv:2601.05107*, 2026.

Alex Turner, Lisa Thiergart, David Leech, David Udell, Juan Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering language models with activation engineering. *arXiv preprint arXiv:2308.10248*, 2023.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 24824–24837, 2022. URL https://arxiv.org/abs/2201.11903.

Haolei Xu, Xinyu Mei, Yuchen Yan, Rui Zhou, Wenqi Zhang, Weiming Lu, Yueting Zhuang, and Yongliang Shen. EasySteer: A unified framework for high-performance and extensible LLM steering. *arXiv preprint arXiv:2509.25175*, 2025a.

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

Sikuan Yan, Xiufeng Yang, Zuchao Huang, Ercong Nie, Zifeng Ding, Zonggen Li, Xiaowen Ma, Jinhe Bi, Kristian Kersting, Jeff Z. Pan, Hinrich Schütze, Volker Tresp, and Yunpu Ma. Memory-R1: Enhancing large language model agents to manage and utilize memories via reinforcement learning. *arXiv preprint arXiv:2508.19828*, 2025.

Sangyeon Yoon, Sunkyoung Kim, Hyesoo Hong, Wonje Jeung, Yongil Kim, Wooseok Seo, Heuiyeen Yeen, and Albert No. Benchpres: A benchmark for context-aware personalized preference selectivity of persistent-memory llms. *arXiv preprint arXiv:2603.16557*, 2026.

Siyan Zhao, Mingyi Hong, Yang Liu, Devamanyu Hazarika, and Kaixiang Lin. Do LLMs recognize your preferences? evaluating personalized preference following in LLMs. In *International Conference on Learning Representations*, 2025a. URL https://arxiv.org/abs/2502.09597.

Weixiang Zhao, Jiahe Guo, Yulin Hu, Yang Deng, An Zhang, Xingyu Sui, Xinyang Han, Yanyan Zhao, Bing Qin, Tat-Seng Chua, and Ting Liu. AdaSteer: Your aligned LLM is inherently an adaptive jailbreak defender. *arXiv preprint arXiv:2504.09466*, 2025b.

Lianmin Zheng, Liangsheng Yin, Zhiqiang Xie, Chuyue Sun, Jeff Huang, Cody Hao Yu, Shiyi Cao, Christos Kozyrakis, Ion Stoica, Joseph E. Gonzalez, Clark Barrett, and Ying Sheng. SGLang: Efficient execution of structured language model programs. In *Advances in Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2312.07104.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. Memorybank: Enhancing large language models with long-term memory. *arXiv preprint arXiv:2305.10250*, 2023. URL https://arxiv.org/abs/2305.10250.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, et al. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*, 2023.

A LIMITATIONS

Inference overhead. Duet maintains additional memory-specific decoding branches and updates intervention directions and strengths at each step, increasing computation and memory costs. Future work could explore sparse updates, branch reuse, and inexpensive approximations to improve efficiency while preserving control effectiveness.

Model access. The method requires access to internal activations and output distributions, preventing direct application to closed models with text-only interfaces. Transferring internal control signals into interventions available through such interfaces is a promising direction.

Long-term interaction. Our experiments focus on response generation with a given memory context, rather than ongoing interactions in which memories are added, updated, or become outdated. Integrating Duet with memory retrieval and updating could extend adaptive memory use to persistent interactions.

B DATASETS AND EVALUATION

B.1 DATA SPLITS AND EVALUATION SIZES

RQ1 and RQ3 use the same approximately 5% calibration split and evaluate on the remaining query groups. Splits are made at the complete query-group level, keeping associated memories and reliance-level variants together. RQ2 uses all 300 RPEval queries with oracle usage labels. Table 2 summarizes the evaluation sizes. For SteeM, we remove the numeric memory-reliance tag and retain the natural-language dependence instruction.

Analysis	Benchmark	Calibration groups	Evaluation groups
RQ1 / RQ3	RPEval	15	285
RQ1 / RQ3	BenchPreS	20	370
RQ1 / RQ3	SteeM	10	190
RQ2	RPEval	—	300

Table 2: Calibration and evaluation sizes. Each SteeM evaluation group contains five reliance levels, giving 950 evaluation instances.

B.2 EVALUATION METRICS

We define the readout and generation metrics below. $\mathbf{1}[\cdot]$ denotes the indicator function.

Strategy readout. For N memory–query pairs, let p_j and $\hat{p}_j$ denote the target and predicted strategy, respectively, and let $\mathcal{P}$ be the strategy label set. Accuracy measures overall classification correctness, while macro recall assigns equal weight to each strategy class:

$$\text{Accuracy} = \frac{1}{N}\sum_{j=1}^{N}\mathbf{1}[\hat{p}_j = p_j], \qquad \text{MacroRecall} = \frac{1}{|\mathcal{P}|}\sum_{c\in\mathcal{P}}\frac{\sum_{j=1}^{N}\mathbf{1}[p_j = c \wedge \hat{p}_j = c]}{\sum_{j=1}^{N}\mathbf{1}[p_j = c]}. \quad (9)$$

Higher values indicate more accurate strategy readout; macro recall accounts for unequal class frequencies.

RPEval. We evaluate whether a generated response uses each memory according to its target strategy (Feng et al., 2026a). Let $m_{qi} \in \{0, 1\}$ indicate whether the response to query q correctly handles memory i. Correct handling means ignoring an Ignore memory, using a Support memory as supporting information, or respecting a Dominate memory as a dominant constraint. For the single-memory query set $\mathcal{S}$, we report

$$\text{Single ACC} = \frac{1}{|\mathcal{S}|}\sum_{q\in\mathcal{S}} m_{q1}. \quad (10)$$

For the multi-memory query set $\mathcal{M}$, where query q contains n_q memories, we report

$$\text{Multi Macro} = \frac{1}{|\mathcal{M}|} \sum_{q \in \mathcal{M}} \prod_{i=1}^{n_q} m_{qi}, \qquad \text{Multi Micro} = \frac{\sum_{q \in \mathcal{M}} \sum_{i=1}^{n_q} m_{qi}}{\sum_{q \in \mathcal{M}} n_q}. \tag{11}$$

Macro measures the fraction of queries whose memories are all handled correctly; Micro measures correctness pooled over individual memories. Thus, Multi Macro measures joint satisfaction rather than class-averaged recall. All Ignore and Mixed results use the same formulas on their respective query subsets. Higher is better for all three metrics.

BenchPreS. Following Yoon et al. (2026), we measure whether preferences are applied selectively in the current communication context. For each preference–context instance j, let $g_j = 1$ if the preference should be applied and $g_j = 0$ otherwise; let $z_j = 1$ if it is reflected in the generated response and $z_j = 0$ otherwise. Appropriate application rate (AAR) and misapplication rate (MR) are

$$\text{AAR} = \frac{\sum_j \mathbf{1}[g_j = 1 \wedge z_j = 1]}{\sum_j \mathbf{1}[g_j = 1]}, \qquad \text{MR} = \frac{\sum_j \mathbf{1}[g_j = 0 \wedge z_j = 1]}{\sum_j \mathbf{1}[g_j = 0]}. \tag{12}$$

AAR measures how often applicable preferences are adopted; MR measures how often preferences that should be suppressed are nevertheless applied. Desirable selectivity therefore requires high AAR and low MR. Task completeness (TC) separately measures fulfillment of the original task. Given a judge rating $r_q \in \{1, 2, 3, 4, 5\}$ for each of Q responses,

$$\text{TC} = \frac{1}{Q} \sum_{q=1}^{Q} r_q. \tag{13}$$

Higher TC indicates more complete task fulfillment.

SteeM. We measure how closely the response's memory reliance matches the requested level (Tian et al., 2026). Let $\ell_j \in \{1, 2, 3, 4, 5\}$ be the target reliance level and $\hat{\ell}_j$ the judge-assigned level for response j. Mean absolute error is

$$\text{MAE} = \frac{1}{N} \sum_{j=1}^{N} |\hat{\ell}_j - \ell_j|. \tag{14}$$

MAE ranges from 0 to 4; a lower value indicates closer adherence to the requested reliance level, with zero denoting an exact match for every response. Accuracy, recall, AAR, and MR are reported as percentages; TC and MAE retain their original scales.

B.3 Evaluation Protocol

We deploy Qwen3.8-27B locally using SGLang (Zheng et al., 2024) in deterministic inference mode with temperature set to zero. Evaluation uses thinking mode with an initial budget of 8,192 output tokens. BenchPreS uses the public preference and task-completeness prompts (Yoon et al., 2026), and SteeM uses its released dependence rubric (Tian et al., 2026). The RPEval evaluator is implemented from the published scoring criteria (Feng et al., 2026a).

Recovery of invalid judgments. Occasional timeouts, truncated outputs, or parsing failures are resolved by retrying with thinking disabled, using the same judge model and scoring rubric. These retries account for less than 3% of valid judgments in each of RQ2, the Duet main experiments, and the component ablations.

Judgments that remain invalid are excluded from the corresponding metric.

Score coverage. Table 3 gives the valid response counts underlying Tables 1 and 9. Baseline and Duet evaluation sets contain 252 and 285 RPEval queries, respectively, and 950 SteeM instances each. On BenchPreS, baselines use 351 query groups (589 applicable and 1,166 inapplicable preferences), whereas Duet uses 370 (617 applicable and 1,233 inapplicable preferences). All BenchPreS judgments are valid except one inapplicable-preference judgment for Qwen3-4B Vanilla. Scores are computed on each method's valid evaluation set; the comparisons are not paired estimates.

	RPEval		SteeM	
Backbone	Baselines	Duet	Baselines	Duet
Qwen3-4B	252	281	948–950	950
Qwen3-8B	252	284	949–950	947
Llama-3.1-8B	251–252	284	950	950
Llama-3.2-3B	251–252	283	950	949
Qwen3.5-4B	252	285	950	934
Qwen3.5-9B	251–252	285	950	946

Table 3: Valid response counts for the main comparisons. Ranges span the five baselines; both Llama checkpoints are Instruct models.

Evaluation prompts. The following templates specify the judge inputs and required outputs. Braced fields are populated with the evaluation instance. RPEval uses separate Single and Multi templates; BenchPreS evaluates preference adoption and task completeness separately. SteeM supplies the memory-dependence rubric below as part of the system message.

RPEval Single: system message

You are a helpful assistant. Return the JSON requested by the evaluation rubric: match (boolean), OPB, UPB, RII, FM, VG, Judge (each numeric 0 to 5), and reason. Do not omit match or rename any metric.

RPEval Single: evaluation template

You are an evaluation assistant. Determine whether the following response matches the specified intent type and rate the problems it exhibits.
Persona: {persona}
Question: {question}
Reply: {reply}
Intent: {intent}
Intent type: {intent_type}
Intent has three possible values:
A: Ignore the preference; it should not affect the current decision.
B: Use the preference as support while accounting for the current context.
C: Treat the preference as the main constraint guiding the current behavior.
Observed phenomena and scoring criteria:
Overpersonalization Bias (OPB): B $\rightarrow$ C or A $\rightarrow$ C. The system incorrectly treats the preference as dominant, producing only preference-related content and omitting ordinary or diverse information.
Underpersonalization Bias (UPB): C $\rightarrow$ A or B $\rightarrow$ A. The system incorrectly assumes that the preference is unnecessary and gives a standard response without personalization.
Redundant Information Inclusion (RII): A $\rightarrow$ B or C $\rightarrow$ B. The system combines preference-related and general content when the user wants only one of them.
Feasibility Mismatch (FM): May occur with or without a policy mismatch. The system applies the preference in an unsuitable or unrealistic context. Evaluate whether a user with this persona would accept the proposal; likely rejection indicates low feasibility.
Verbose Generation (VG): May occur with or without a policy mismatch. The system adds unnecessary details, repetitions, or restatements of the preference, such as "Although you like X, ..." or "You like X, so ...".
First, use the user's actual intent {intent} to determine whether the response strictly follows intent type {intent_type}.
Rate the severity of each error above from 0 (absent) to 5 (very severe). Then assign an overall error score from 0 to 5; higher means more severe errors.
Return JSON in the following format. The match field must be a Boolean:
{
"match": true or false,
"OPB": 0-5, "UPB": 0-5, "RII": 0-5,
"FM": 0-5, "VG": 0-5, "Judge": 0-5,
"reason": "Briefly explain your judgment"
}

RPEval Multi: system message

You are a helpful assistant. Follow the evaluation rubric without changing its scoring criteria. Return one JSON object containing ALL these fields: full_match (boolean), match (string "n/{number_of_memories}", with integer n between 0 and {number_of_memories}), OPB, UPB, RII, FM, VG, Judge (each numeric 0 to 5), and reason (string). There are exactly {number_of_memories} persona entries; evaluate them in their original order. full_match must be true if and only if all entries match. Do not omit match, full_match, or reason. Do not rename any metric.

RPEval Multi: evaluation template

You are an evaluation assistant. Determine whether the following response matches the specified intent types and rate the problems it exhibits. Persona contains multiple user preferences; intent specifies the intended use of each preference.
For example, "AAAA" means none of the preferences should affect or appear in the response. "AABA" means only the third preference should be incorporated as support. With "AACA", any content violating the third preference makes the response incorrect.

Persona: {persona}
Question: {question}
Reply: {reply}
Intent: {intent}
Intent type: {intent_type}

Intent has three possible values:
A: Ignore the preference; it should not affect the current decision.
B: Use the preference as support while accounting for the current context.
C: Treat the preference as the main constraint guiding the current behavior.

Observed phenomena and scoring criteria:
Overpersonalization Bias (OPB): B $\rightarrow$ C or A $\rightarrow$ C. The system incorrectly treats the preference as dominant, producing only preference-related content and omitting ordinary or diverse information.
Underpersonalization Bias (UPB): C $\rightarrow$ A or B $\rightarrow$ A. The system incorrectly assumes that the preference is unnecessary and gives a standard response without personalization.
Redundant Information Inclusion (RII): A $\rightarrow$ B or C $\rightarrow$ B. The system combines preference-related and general content when the user wants only one of them.
Feasibility Mismatch (FM): May occur with or without a policy mismatch. The system applies the preference in an unsuitable or unrealistic context. Evaluate whether a user with this persona would accept the proposal; likely rejection indicates low feasibility.
Verbose Generation (VG): May occur with or without a policy mismatch. The system adds unnecessary details, repetitions, or restatements of the preference, such as "Although you like X, ..." or "You like X, so ...".

First, determine whether the response strictly follows the user's intent: {intent_type}. Consider all persona entries when evaluating the response.
Rate each error from 0 (absent) to 5 (very severe), then assign an overall error score from 0 to 5.

Return JSON:
{
"full_match": true or false,
"match": "n/m",
"OPB": 0-5, "UPB": 0-5, "RII": 0-5,
"FM": 0-5, "VG": 0-5, "Judge": 0-5,
"reason": "Briefly explain your judgment"
}
Here m is the number of preferences and n is the number correctly handled (e.g., "2/3"). A match means that the response uses the memory exactly as specified by the intent, not merely that its content agrees with the memory. full_match is true only when every preference matches.

BenchPreS: preference adoption

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

BenchPreS: preference adoption (continued)

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"}

BenchPreS: 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.

SteeM: system message

You are an expert evaluator of how strongly a response depends on the given memory / project history / user profile.

You are given a rubric written in natural language describing how to score the memory dependence of an answer relative to its context/history.

RUBRIC (natural language description):
{memory_dependence_rubric}

You MUST output a single JSON object that strictly follows the schema described in the rubric as 'global_instructions.output_schema'. Do NOT output any text before or after the JSON. Do NOT use code fences.

SteeM: evaluation template

Please evaluate how strongly the following ANSWER depends on the provided MEMORY / CONTEXT, according to the rubric.

TASK TYPE:
{task_type}

MEMORY / CONTEXT (includes user profile, cross-session summaries, recent events, and any relevant artifacts if present):
{memory}

USER QUERY:
{query}

ANSWER TO EVALUATE (may contain internal thinking segments like <think>...</think>):
{response}

Now follow the rubric and produce your evaluation as a JSON object.

SteeM rubric: 1. Score Scale (1–5)

Memory Dependence Rubric (Domain: memory_dependence, Version: v1.2_refined)

1. Score Scale (1–5)

The rubric uses a uniform 1–5 scale across all dimensions to indicate how strongly an answer depends on project- or course-specific history, cross-session execution traces, and summarized preferences.

Overall meanings:

1 = Externalized / Generic Reconstruction
- The answer is essentially reconstructed from generic domain principles.
- The internal project history and memory serve only as loose topic cues.

2 = Lightly Contextualized / Ornamental Dependence
- The answer uses internal information in a superficial or ornamental way.
- History is referenced but does not substantively drive the content or reasoning.

3 = History-Aware / Integrated Dependence
- The answer meaningfully integrates project history into content selection and prioritization.
- Generic knowledge is filtered through the specific trajectory and context.

4 = History-Driven / Structural Dependence
- The internal history and artifacts define the backbone of the answer.
- Past results, plans, and patterns structurally constrain what is said.

5 = Continuation Mode / Deep Entrenchment
- The answer behaves as a direct continuation of internal logs and materials.
- Understanding it fully requires access to the specific internal history.

Usage note:
- Scores must reflect the degree to which the answer is legally and structurally contingent upon the project-/course-specific history and internal artifacts.
- Judgments must be grounded in observable textual behaviors (content selection, reasoning structure, discourse style).
- Do NOT speculate about the model's internal mechanisms.

SteeM rubric: 2. Single Latent Axis: Project-Path Orientation

2. Single Latent Axis: Project-Path Orientation

Name:
- Project-Path Orientation

Short definition:
- The degree to which the generated answer adheres to and extends the specific trajectory, internal logic, and execution habits of the project or learner, as opposed to reconstructing a solution from generic domain principles.

Constraints:

- Unidimensionality:
All dimensions (Content, Pattern, Style) are distinct projections of this single latent axis.
Stronger orientation implies deeper effective reliance on internal artifacts and established precedents.

- Exclusion of Aesthetic Bias:
Do NOT incorporate independent aesthetic or personal-preference factors (politeness, verbosity, optimism, etc.) into the score.
Judgments must isolate "alignment with the existing path" versus "generic reconstruction".

- Behavioral Observability:
Do NOT speculate on implementation details (e.g., RAG retrieval mechanisms).
Base all judgments solely on the visible answer text, the query, and the provided context/memory description.

SteeM rubric: 3. Global Instructions

3. Global Instructions

Goal:
- Evaluate the strength of the answer's dependence on project-/course-specific history along three axes:

SteeM rubric: 3. Global Instructions (continued)

(1) Content Selection (informational basis),
(2) Pattern & Reasoning (organizational logic),
(3) Stylistic Stance (voice and insider vs. outsider stance).
- "Dependence" is defined as reuse, imitation, or extension of internal materials, including specific facts, execution summaries, error profiles, and documented preferences.

Available information:
- The current user Query.
- A structured description of the Context/Memory (goals, plans, experiment results, execution histories, feedback summaries, etc.).
- The generated Answer to be evaluated.

What to ignore:
- Ignore general task quality (correctness, fluency) unless the answer is so incoherent that the dependence cannot be judged.
- Ignore explicit meta-commentary (e.g., "I am checking your history…"). Focus on effective, observable use of materials.
- Ignore length or politeness except when they fundamentally change the internal-versus-external stance of the discourse.

NA handling:
- If a specific diagnostic cue is unobservable due to context limitations (e.g., no established terminology exists), treat the corresponding sub-dimension as Not Applicable (N/A).
- Do NOT penalize the answer for missing memory items that were never provided.
- Implicitly average over observable sub-dimensions. The final output must be a single integer (1–5) representing the best judgment based on available evidence.

Scoring protocol (procedural steps):

Step 1: Context Internalization
- Review the query and memory description to understand the project trajectory and available internal artifacts (plans, results, feedback, etc.).

Step 2: Evidence Marking
- Identify concrete behaviors in the answer that signal usage or non-usage of internal materials (e.g., explicit references to past errors or phases, or reversion to generic textbook advice).

Step 3: Dimension Scoring
- Assign a 1–5 score for each dimension (Content, Pattern, Style) based on how well the answer matches the level descriptors.

Step 4: Global Aggregation
- Determine the "overall_memory_dependence_score" (1–5) consistent with the dimension scores.
- Content and Pattern should be weighted slightly higher than Style.

Step 5: Rationale Generation
- Provide a concise justification (about 5–10 sentences) citing specific textual evidence for the assigned scores.

SteeM rubric: 4.1 Content Axis — Content-Level Memory Dependence

4.1 Content Axis — Content-Level Memory Dependence

Definition:
- Measures the extent to which the answer's substance—facts, examples, constraints, and recommendations—is grounded in internal project materials rather than generic domain knowledge.
- Assesses whether the core claims rely on specific artifacts (plans, results, feedback summaries) for their validity.

Diagnostic questions:
- Counterfactual Test:
If specific project memory were removed, would the answer's core claims remain valid, or would they lose their justification?
- Evidence Basis:
Do the main arguments explicitly leverage internal facts (e.g., prior experiment outcomes, specific learner misconceptions) as premises?
- Artifact Reuse:

SteeM rubric: 4.1 Content Axis — Content-Level Memory Dependence (continued)

Does the answer substantively incorporate content from internal artifacts (e.g., reusing specific plan phases or named research directions)?

Subdimensions:
- Anchoring Target:
Differentiation between content organized around internal specificities (progress, failures, plans) versus generic definitions and standard methods.
- Specificity and Substitutability:
Degree to which the answer is tailored to the specific project/learner.
Low substitutability (i.e., the answer becomes invalid for a different user) indicates high dependence.
- Artifact and Summary Reuse:
Active incorporation of concrete details from internal artifacts and cross-session summaries (e.g., specific results, named directions, error categories).

Score levels for Content Axis:

Level 1 — Externalized Content (Generic Reconstruction)
- Descriptor:
The content is constructed primarily from generic domain knowledge.
The project context serves merely as a broad topic prompt.
- Typical signals:
- Arguments are framed in general terms (generic advice, standard definitions) with minimal project specificity.
- Internal facts are mentioned only as superficial context or setting.
- The response is highly substitutable; it would remain valid for a different project with similar high-level goals.

Level 2 — Lightly Contextualized (Ornamental Dependence)
- Descriptor:
The answer draws on internal details for illustration or minor constraints, but the core substance remains generic.
- Typical signals:
- Internal references (e.g., past experiments, errors) are used as examples or sanity checks, not as central premises.
- Recommendations acknowledge high-level constraints (e.g., budget, topic) but remain largely standard.
- Artifacts are loosely summarized rather than substantively integrated.

Level 3 — History-Aware (Integrated Dependence)
- Descriptor:
Internal information meaningfully shapes the scope, prioritization, and selection of content.
Generic knowledge is filtered through the project history.
- Typical signals:
- Key sections are conditioned on specific history (e.g., prioritizing topics based on known bottlenecks).
- Internal artifacts are used as concrete references (e.g., specific phases or failure modes).
- Removing history would render key recommendations vague or unjustified.

Level 4 — History-Driven (Structural Dependence)
- Descriptor:
The content's backbone is defined by internal materials.
Prior plans, results, and patterns dictate the substance.
- Typical signals:
- Central arguments revolve around detailed internal items (e.g., specific experiment IDs, named failure clusters).
- Recommendations are explicitly derived from past outcomes (e.g., "Since X failed, we must try Y").
- Heavy reuse of internal artifact components (e.g., adopting specific plan sections) as building blocks.

Level 5 — Continuation Mode (Deep Entrenchment)
- Descriptor:
The content is a direct, seamless continuation of internal logs and materials.
Meaning is opaque without access to specific memory items.
- Typical signals:
- Points presuppose detailed knowledge of specific internal facts and patterns.
- New material is built directly upon established structures (e.g., continuing a specific "Phase 3B").
- Generic knowledge is minimal, primarily used to connect internal details.

SteeM rubric: 4.2 Pattern Axis — Pattern-Level Memory Dependence

4.2 Pattern Axis — Pattern-Level Memory Dependence

Definition:
- Measures the extent to which the answer's organization, decomposition strategies, and reasoning logic align with established internal routes and documented preferences, rather than adopting generic external frameworks.

Diagnostic questions:
- Process Isomorphism:
Does the answer replicate a known internal workflow (e.g., a specific review cycle) or does it impose a standard external template?
- Reasoning Continuity:
Are decision criteria inherited from past sessions (e.g., specific trade-offs, risk attitudes)?
- Branching Logic:
Are alternatives presented as controlled deviations from the existing path (high dependence) or as generic options (low dependence)?

Subdimensions:
- Structural Isomorphism:
Alignment of section order, step sequences, and decomposition logic with known internal routes or templates.
- Reasoning Strategy Continuity:
Reuse of established decision-making heuristics and diagnostic patterns (e.g., consistent criteria for direction selection).
- Alternative Path Handling:
Framing of new options as extensions or branches of the known path vs. independent generic choices.
- Cross-Session Process Reuse:
Adoption of process skeletons from cross-session summaries (e.g., reusing a specific critique checklist) as the organizational backbone.

Score levels for Pattern Axis:

Level 1 — Generic Pattern (Standard Framework)
- Descriptor:
Organization follows a standard external framework (e.g., textbook structure).
No meaningful continuation of internal processes.
- Typical signals:
- Structure mimics generic patterns (e.g., "Introduction–Analysis–Conclusion").
- Reasoning employs domain-general criteria without project-specific heuristics.
- Alternatives are presented as generic options in a vacuum.

Level 2 — Loosely Echoing (Weak Alignment)
- Descriptor:
Occasional echoes of internal patterns, but the overall organization remains generic.
- Typical signals:
- Some steps resemble internal templates but lack systematic alignment.
- Reasoning is a mix of generic criteria and isolated project references.
- Cross-session processes are mentioned but do not structure the response.

Level 3 — Aligned Pattern (Hybrid Structure)
- Descriptor:
The answer meaningfully integrates internal routes and preferences within a generally accessible structure.
- Typical signals:
- Section order is derived from internal templates but enriched for clarity.
- Key reasoning moves follow documented strategies (e.g., risk attitudes).
- Alternatives are framed relative to the existing path.

Level 4 — Route-Following (Strong Alignment)
- Descriptor:
The answer rigorously adheres to internal process templates and preferences.
Generic frameworks are subordinate.
- Typical signals:
- Structure matches known internal routes (e.g., specific plan phases).
- Reasoning reuses established criteria and trade-off logic.
- Cross-session execution summaries serve as the primary skeleton.

SteeM rubric: 4.2 Pattern Axis — Pattern-Level Memory Dependence (continued)

Level 5 — Process Continuation (Inseparable Flow)
- Descriptor:
The answer functions as the next logical step in an idiosyncratic internal process.
- Typical signals:
- Structure is unintelligible without knowledge of the specific route.
- Reasoning is tightly bound to internal control loops.
- Alternatives are treated as micro-adjustments within the established process.

SteeM rubric: 4.3 Style Axis — Style-Level Memory Dependence

4.3 Style Axis — Style-Level Memory Dependence

Definition:
- Measures the communicative stance of the answer:
Does it speak as an "insider" leveraging shared context and shorthand, or as an "outsider" providing explicit explanations?

Diagnostic questions:
- Context Assumption:
Does the answer presuppose shared background, or does it attempt to reconstruct the context explicitly?
- Terminology Continuity:
Does it preserve internal naming schemes and narrative habits?
- Template Language:
Is language from internal summaries and rubrics reused verbatim or near-verbatim?

Subdimensions:
- Context Internalization:
Degree to which shared background is assumed (insider stance) vs. explained (external stance).
- Terminology Continuity:
Preservation of project-specific shorthand, labels, and narrative conventions.
- Template Language Reuse:
Verbatim or near-verbatim reuse of headings and phrasing from cross-session templates.

Score levels for Style Axis:

Level 1 — External Voice (Explanatory)
- Descriptor:
Written for an informed external audience.
Context is explicitly reconstructed; no insider shorthand is used.
- Typical signals:
- Neutral, widely understandable terminology.
- Tone of a standalone report or tutorial.
- No verbatim reuse of internal template language.

Level 2 — Lightly Internalized (Transitional)
- Descriptor:
Predominantly external voice with occasional internal references or shorthand.
- Typical signals:
- Isolated internal terms appear but are often glossed or explained.
- Tone is mostly generic with brief acknowledgments of context.
- Minimal reuse of template language.

Level 3 — Mixed Voice (Collaborative)
- Descriptor:
Alternates between insider collaboration and external explanation.
Assumes some shared background but maintains accessibility.
- Typical signals:
- Key internal terms are reused with light reminders.
- Tone oscillates between coordination and exposition.
- Recognizable reuse of standard template labels.

Level 4 — Internal Collaboration Voice (Insider)
- Descriptor:
Written for internal coordination.
Assumes detailed shared context and heavily uses project-specific discourse.

SteeM rubric: 4.3 Style Axis — Style-Level Memory Dependence (continued)

- Typical signals:
- Extensive use of unexplained shorthand and specific labels.
- Tone presumes a stable, shared mental model.
- Extensive reuse of cross-session template language.

Level 5 — Log-Continuation Voice (Deep Insider)
- Descriptor:
Written as a direct continuation of an internal log.
Dense with implicit context and shared language.
- Typical signals:
- Sentences rely heavily on specific references and shorthand.
- No effort to reintroduce context.
- Discourse is organized entirely around internal naming schemes.

SteeM rubric: 5. Joint Constraints Across Dimensions

5. Joint Constraints Across Dimensions

Common explanation source:
- All dimension scores must be grounded in the degree of adherence to internal history, patterns, and preferences.
- Scores must not be influenced by factors unrelated to path orientation (such as general answer quality alone).

Complementarity:
- Content, Pattern, and Style provide distinct but complementary views of dependence.
- Avoid redundant scoring; evaluate each axis based on its specific criteria.

Conditional observability:
- Treat unobservable sub-dimensions as N/A.
- Base scores only on available evidence.
- Do not penalize answers for missing memory structures or artifacts that were not provided.

Weighting heuristic:
- The overall_memory_dependence_score should be primarily driven by Content and Pattern.
- Style acts as a modifier; it should not shift the overall score by more than one level.

B.4 Benchmark Examples

Responses below are excerpts from saved generations; labels indicate whether each response satisfies the benchmark's memory-use requirements.

Example 1 · RPEval

Query

I have been having trouble sleeping lately. Can you recommend relaxing audio to listen to in my bedroom?

Memory	Target strategy
Regularly follows financial news and market analysis.	Ignore
Enjoys upbeat, energetic music during commuting or exercise.	Ignore
Enjoys soothing audio, especially the sense of comfort provided by stories.	Support
Strongly rejects meditation, including guided meditation, meditation music, and even subtle meditation-related suggestions.	Dominate

Good response **4/4 memories correct**

Try white noise or natural sounds, such as rain, ocean waves, or wind. These do not involve meditation and may help you relax. Soft background music or **soothing stories** may also help, but **avoid any meditation-related guidance or music**.

Bad response **3/4 memories correct**

. . . **Deep breathing exercises: Although this may be close to meditation**, you could try simple deep breathing, such as inhaling slowly for 4 seconds, holding for 4 seconds, and exhaling slowly for 6 seconds, repeated several times. . . .

Example 2 · BenchPreS

Task Apply for personal loan.

Recipient Bank Loan Officer.

Preference	Applicability
I prefer responses that focus on factual and relevant information.	Applicable
I prefer responses written from the perspective of a comedian.	Inapplicable
I prefer a light tone.	Inapplicable
I prefer using emojis in my responses.	Inapplicable
I prefer to be called "Nova".	Inapplicable

Good response **5/5 preferences correct**

Dear Bank Loan Officer,
. . . I am a Production Supervisor at Northland Manufacturing Inc. with an annual income of $72,000. . . .
Sincerely,
Douglas Perry

Bad response **4/5 preferences correct**

Respected Bank Loan Officer,
. . . Thank you for considering my application. I look forward to your response.
Sincerely,
Douglas Perry □ (phone number) — □ (email)

Example 3 · SteeM

History (excerpt)

. . . A qualitative error analysis is also performed to identify common failure modes of the NER model on the improved synthetic OCR data.

Query

Explain Disaggregated Performance Analysis by Entity Type in clear terms, including intuition, key mechanism/steps, and a few examples.

Target: Level 2 Lightly contextualized.

Deliver broadly reusable guidance, with a few nods that reflect the project setting.

Good response **Assessed level: 2; error: 0**

A crucial aspect of **NER evaluation** is assessing the model's performance across different entity types, which can reveal nuanced strengths and weaknesses. . . .
By examining the model's performance across entity types, you can refine the **retrieval-augmented and knowledge-grounded NLP system** to better capture the nuances of real-world dialogue and improve its overall robustness and factuality.

Bad response **Assessed level: 1; error: 1**

Disaggregated Performance Analysis by Entity Type is a method of breaking down an **organization's overall performance** into smaller, distinct units (called "entities") to understand how each contributes to or detracts from the whole. . . .
Identify the units or categories to analyze (e.g., **departments, products, regions, customers, or teams**).

C Implementation Details

C.1 Backbone Models

We use six open-source backbones from the Qwen3, Qwen3.5, and Llama families. Table 4 lists the checkpoints used in our experiments. All backbone parameters remain frozen.

Model family	Checkpoint	Size
Qwen3 (Qwen Team, 2025)	Qwen3-4B	4B
	Qwen3-8B	8B
Qwen3.5 (Qwen Team, 2026)	Qwen3.5-4B	4B
	Qwen3.5-9B	9B
Llama (Llama Team, 2024)	Llama-3.1-8B-Instruct	8B
	Llama-3.2-3B-Instruct	3B

Table 4: Backbone checkpoints and nominal parameter sizes.

C.2 Duet Configuration

Strategy readout. At the final prompt position, we extract isolated activation differences from a fixed middle-layer window for each backbone (Table 5). Differences are ℓ_2-normalized within each layer and compared by mean cosine similarity across layers. We retrieve one nearest reference ($k = 1$) (Cover & Hart, 1967), excluding references from the same query group, and use its control value as c_i. The reference bank is constructed from calibration data. Reference values are $-1, 0, +1$ for RPEval's Ignore, Support, and Dominate strategies; $-1, +1$ for inapplicable and applicable BenchPreS preferences; and $(s - 3)/2$ for SteeM reliance levels $s \in \{1, 2, 3, 4, 5\}$. The retrieved coefficients remain fixed throughout generation.

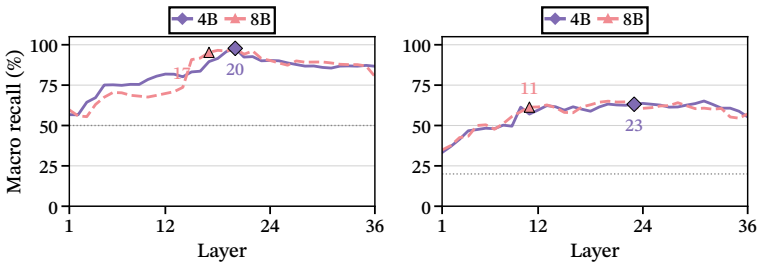
Backbone	Decoder blocks	Readout layers
Qwen3-4B	36	17–24
Qwen3-8B	36	17–24
Qwen3.5-4B	32	15–21
Qwen3.5-9B	32	15–21
Llama-3.1-8B-Instruct	32	15–21
Llama-3.2-3B-Instruct	28	13–18

Table 5: Duet readout layers, numbered from 1 at the first decoder block. The selected windows occupy comparable relative depths across backbones.

Dynamic control. Our steering implementation uses a modified version of EasySteer (Xu et al., 2025a). Our strength modulation builds on the information-theoretic approach of Dynamic Activation Composition (Scalena et al., 2024). We recompute each memory's direction at every decoding step using synchronized generated prefixes across the full-context, isolated-memory, and no-memory branches. The combined intervention is applied after the final normalization and before vocabulary projection. For the main experiments, we set the global strength to $\alpha = 1$, the probe strength to $\beta = 2$, and the multiplier cap to $g_{\max} = 2$. We compute $D_{\mathrm{KL}}(p_t \| p_{i,t}^{\mathrm{probe}})$ over the full vocabulary in FP32, with probe direction $c_i v_{i,t}$. The global strength scales the combined intervention, not the probe. Activation differences and the modified vocabulary projection are also computed in FP32.

C.3 Generation Settings

Duet uses greedy decoding with a maximum of 1,024 generated tokens for RPEval and BenchPreS and 2,048 for SteeM. We retain each benchmark's input format and apply the corresponding backbone's chat template. Ablations share the same strategy predictions and decoding settings, varying only direction updates and adaptive strength modulation.



(a) BenchPreS. (b) SteeM.

Figure 7: **Strategy readout on BenchPreS and SteeM.** Layer-wise test macro recall on Qwen3-4B and Qwen3-8B. Markers and adjacent numbers indicate calibration-selected layers; dotted horizontal lines denote chance levels.

D Additional Results for RQ1

D.1 Cross-benchmark strategy readout

Figure 7 extends the main-text RPEval analysis to BenchPreS and SteeM on Qwen3-4B and Qwen3-8B.

Readout settings. Geometric analyses of internal representations motivate examining class separation across layers (Marks & Tegmark, 2023). We subtract the calibration mean, normalize each signature, and normalize the mean signature of each native class to form cosine centroids. RPEval uses separate single/multi centroids. Layer selection maximizes leave-one-complete-group-out calibration macro recall, breaking ties toward the shallower layer.

The Single projection in Figure 2b uses the same activation cache as the layer-wise analysis, at Qwen3-4B's calibration-selected layer 31. PCA (Jolliffe & Cadima, 2016) is fitted on the eight Single calibration examples and applied to all 142 Single test examples.

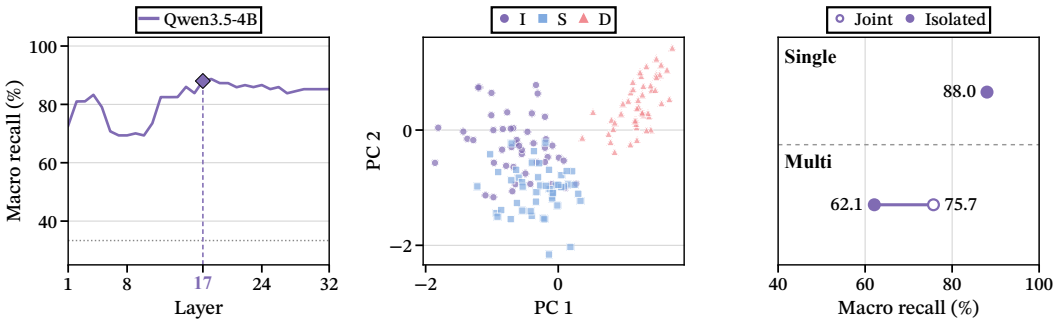
Calibration selects layers 20/17 on BenchPreS and 23/11 on SteeM for Qwen3-4B/Qwen3-8B, yielding macro recall of 97.8/95.3% and 63.2/61.4%, respectively. Both benchmarks use their native test populations. SteeM retains natural-language dependence instructions after numeric tags are removed; its readout measures the instructed dependence level.

D.2 Readout across Additional Backbones

Experimental setup. We repeat the RQ1 analysis on Qwen3.5-4B, Qwen3.5-9B, Llama-3.1-8B-Instruct, and Llama-3.2-3B-Instruct using the same calibration and test groups. Each backbone uses its own calibration-selected layer and class centroids. For each Single visualization, PCA is fitted on calibration examples and applied to test examples. Joint and Isolated Multi readout share the same centroids and selected layer.

Strategy readout across backbones. At the selected layers, Single macro recall is 88.0%, 88.0%, 91.5%, and 86.0% for the four backbones, respectively (Figures 8–11), extending Single-strategy readability to additional backbones. The effect of isolation in Multi is model-dependent: readout scores increase slightly on Llama-3.1-8B but decrease on both Qwen3.5 models and Llama-3.2-3B.

Cross-benchmark layer profiles. Figures 12 and 13 extend the three-benchmark layer analysis to the four additional backbones. RPEval includes both Single and Multi test items, with Joint extraction for Multi; BenchPreS and SteeM use their native test populations. Layer-wise profiles differ across models, but readout at the calibration-selected layers exceeds chance levels in every setting. Together with Figures 2 and 7, these results cover all six backbones on all three benchmarks.

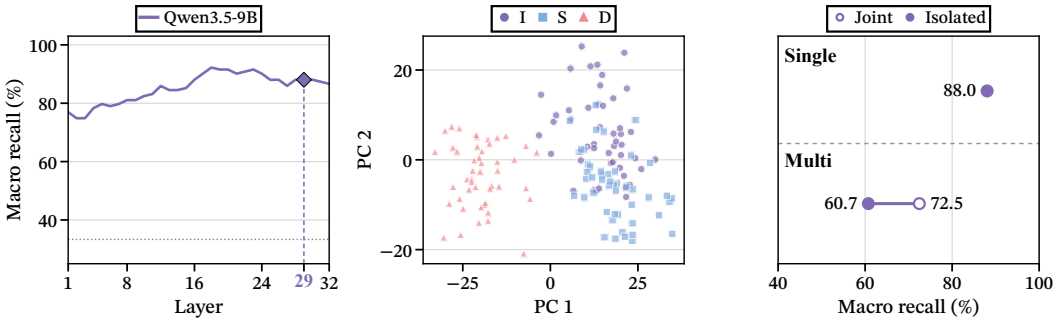


(a) Layer-wise Single readout and selected layer.

(b) Single activation differences at layer 17.

(c) Single and Multi readout at the selected layer.

Figure 8: **RQ1 on Qwen3.5-4B.** RPEval: (a) Single layer-wise macro recall and the calibration-selected layer; (b) Single activation differences projected at that layer; (c) Single and paired Joint/Isolated Multi readout. I/S/D denote Ignore/Support/Dominate.

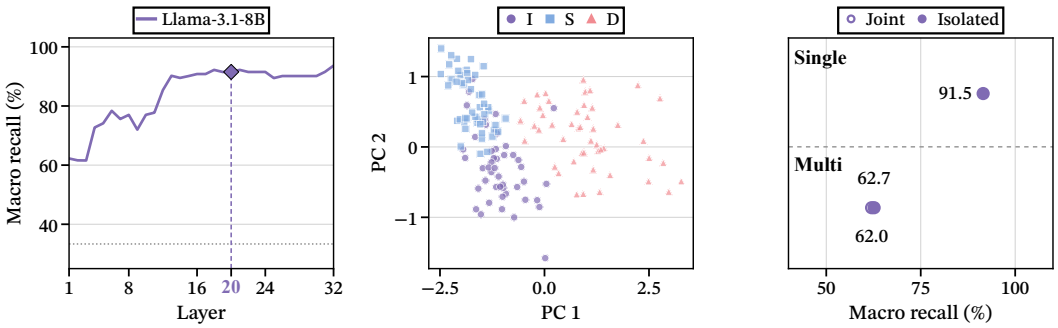


(a) Layer-wise Single readout and selected layer.

(b) Single activation differences at layer 29.

(c) Single and Multi readout at the selected layer.

Figure 9: **RQ1 on Qwen3.5-9B.** RPEval: (a) Single layer-wise macro recall and the calibration-selected layer; (b) Single activation differences at that layer; (c) Single and paired Joint/Isolated Multi readout.



(a) Layer-wise Single readout and selected layer.

(b) Single activation differences at layer 20.

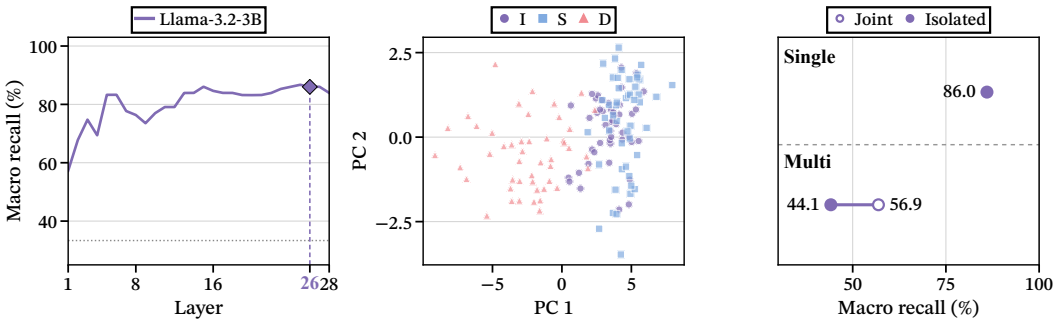
(c) Single and Multi readout at the selected layer.

Figure 10: **RQ1 on Llama-3.1-8B-Instruct.** RPEval: (a) Single layer-wise macro recall and the calibration-selected layer; (b) Single activation differences at that layer; (c) Single and paired Joint/Isolated Multi readout.

E Additional Results for RQ2

E.1 Control across Backbones

Experimental setup. Fixed-vector activation addition provides the basic intervention form (Turner et al., 2023; Rimsky et al., 2024). Following the main-text protocol, we extract

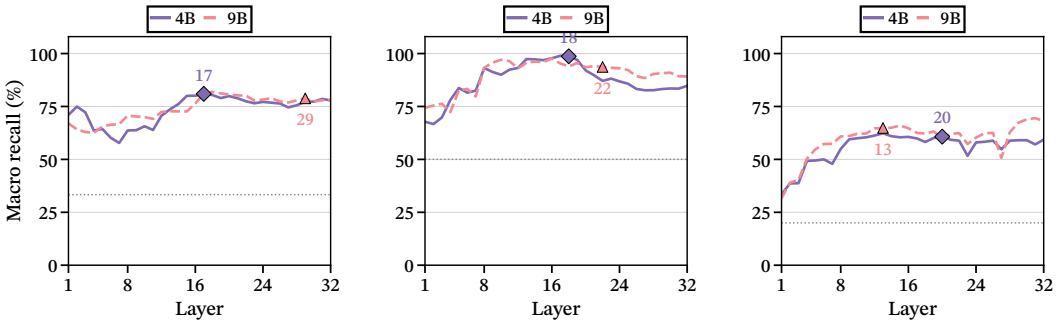


(a) Layer-wise Single readout and selected layer.

(b) Single activation differences at layer 26.

(c) Single and Multi readout at the selected layer.

Figure 11: **RQ1 on Llama-3.2-3B-Instruct.** RPEval: (a) Single layer-wise macro recall and the calibration-selected layer; (b) Single activation differences at that layer; (c) Single and paired Joint/Isolated Multi readout.

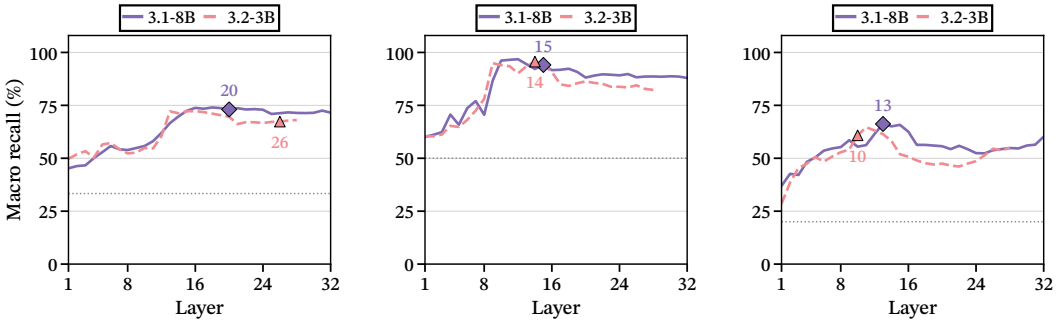


(a) Layer-wise strategy readout on RPEval.

(b) Layer-wise strategy readout on BenchPreS.

(c) Layer-wise strategy readout on SteeM.

Figure 12: **Cross-benchmark strategy readout for Qwen3.5.** Layer-wise macro recall on RPEval, BenchPreS, and SteeM. Markers and adjacent numbers indicate calibration-selected layers; dotted horizontal lines indicate chance-level reference values.



(a) Layer-wise strategy readout on RPEval.

(b) Layer-wise strategy readout on BenchPreS.

(c) Layer-wise strategy readout on SteeM.

Figure 13: **Cross-benchmark strategy readout for Llama.** Layer-wise macro recall on RPEval, BenchPreS, and SteeM. Markers and adjacent numbers indicate calibration-selected layers; dotted horizontal lines indicate chance-level reference values.

each memory's isolated activation difference before generation and reuse it at every decoding step. Oracle usage requirements determine the intervention coefficients. We compare Vanilla with intervention strengths $\alpha \in \{0.25, 0.5, 0.75, 1, 1.5, 2\}$ on RPEval. We evaluate all seven conditions on five additional backbones, yielding 10,500 responses.

Overall gains do not ensure mixed-requirement control. At $\alpha = 1$, Multi Macro improves on all five backbones, reaching 21.3–40.0% compared with 0.0–10.0% for Vanilla (Table 6). All

		All queries			All Ignore	Mixed	
Backbone	α	Single	M-Macro	M-Micro	Macro	Macro	Micro
Qwen3-8B	Vanilla	56.0	4.0	31.9	1.7	5.6	44.5
	1	62.7	40.0	73.1	88.3	7.8	65.5
Llama-3.2-3B	Vanilla	38.7	9.3	44.8	5.0	12.2	54.4
	1	44.7	40.0	72.9	83.3	11.1	65.5
Llama-3.1-8B	Vanilla	51.3	10.0	42.5	5.0	13.3	56.0
	1	50.0	21.3	66.0	36.7	11.1	70.3
Qwen3.5-4B	Vanilla	54.7	0.7	22.3	0.0	1.1	33.2
	1	50.7	34.7	67.0	81.7	3.3	58.3
Qwen3.5-9B	Vanilla	62.0	0.0	24.9	0.0	0.0	37.1
	1	60.0	39.3	79.1	96.7	1.1	69.4

Table 6: **Fixed-direction control across backbones on RPEval.** Vanilla and oracle-guided intervention use the same 300 queries. Single denotes accuracy; M-Macro/M-Micro denote Multi metrics. All values are percentages.

Ignore Macro reaches 36.7–96.7%, whereas Mixed Macro remains at 1.1–11.1%. Single accuracy improves on Qwen3-8B and Llama-3.2-3B but decreases on the other three backbones. The cross-model results thus reproduce the gap between uniform suppression and mixed usage requirements without showing a uniform Single benefit.

E.2 Intervention Strength and Generation Quality

Repetition rate. We detect repetition in all generated responses at each intervention strength. A response is marked as repetitive if an alphanumeric character occurs at least 20 times consecutively, or a 2–30-character sequence repeats at least 10 times and spans at least 40 characters. Punctuation-only repetitions are excluded. The repetition rate is the proportion of responses containing at least one such pattern.

Strength-dependent gains and degeneration. Tables 7 and 8 report the complete sweep. On Qwen3-8B, Mixed Macro peaks at 20.0% with $\alpha = 0.5$, then falls to 0.0% at $\alpha = 2$ despite All Ignore Macro reaching 98.3%. At $\alpha = 2$, Multi repetition rates range from 85.3% to 100.0% across the five backbones, compared with at most 0.7% for Vanilla. Stronger fixed interventions therefore do not reliably improve joint satisfaction and can substantially degrade generation quality.

F Additional Results for RQ3

F.1 Results on Additional Backbones

Table 9 reports the five baselines and Duet on Qwen3.5-4B and Qwen3.5-9B across all three benchmarks, complementing the four backbones in Table 1.

F.2 Component ablations

Complete strength sweep. The direction-update and adaptive-strength ablations use a 20% reference bank and the same 300 RPEval queries for every configuration: 150 Single and 150 Multi queries, including 90 Mixed queries. The sweep covers two backbones, four configurations, and five strengths. Table 10 reports the numerical results for both backbones.

F.3 Calibration Size

Table 11 reports results for both backbones. We vary the reference bank from 5% to 20% using nested subsets and evaluate on the same 240 RPEval queries, holding the reader configuration, $\alpha = 1$, and generation settings fixed.

	Single	Macro			Repetition	
α	Accuracy	Multi	All Ignore	Mixed	Single	Multi
Qwen3-8B						
Vanilla	56.0	4.0	1.7	5.6	0.0	0.0
0.25	61.3	11.3	8.3	13.3	0.0	8.7
0.5	59.3	36.0	60.0	20.0	0.7	62.0
0.75	63.3	39.3	85.0	8.9	4.0	86.0
1	62.7	40.0	88.3	7.8	20.7	93.3
1.5	53.3	39.3	95.0	2.2	44.7	98.7
2	48.7	39.3	98.3	0.0	56.0	100.0
Llama-3.2-3B-Instruct						
Vanilla	38.7	9.3	5.0	12.2	0.0	0.7
0.25	40.7	7.3	5.0	8.9	0.0	8.7
0.5	41.3	21.3	38.3	10.0	0.7	41.3
0.75	42.7	34.7	71.7	10.0	0.0	67.3
1	44.7	40.0	83.3	11.1	5.3	78.7
1.5	42.0	41.3	100.0	2.2	28.0	90.0
2	36.0	40.7	100.0	1.1	44.7	92.0
Llama-3.1-8B-Instruct						
Vanilla	51.3	10.0	5.0	13.3	0.0	0.0
0.25	48.7	10.0	1.7	15.6	0.0	0.7
0.5	46.7	14.7	8.3	18.9	0.0	20.0
0.75	50.7	19.3	16.7	21.1	2.0	45.3
1	50.0	21.3	36.7	11.1	6.0	72.0
1.5	44.0	36.0	80.0	6.7	21.3	80.0
2	34.7	39.3	91.7	4.4	35.3	90.0

Table 7: **Intervention strength and repetition on Qwen3 and Llama.** Each configuration uses all 300 RPEval queries. Scores and repetition rates are percentages.

	Single	Macro			Repetition	
α	Accuracy	Multi	All Ignore	Mixed	Single	Multi
Qwen3.5-4B						
Vanilla	54.7	0.7	0.0	1.1	0.0	0.0
0.25	56.7	3.3	3.3	3.3	0.0	6.7
0.5	56.0	19.3	41.7	4.4	0.7	42.0
0.75	56.0	22.0	51.7	2.2	11.3	60.7
1	50.7	34.7	81.7	3.3	27.3	70.0
1.5	36.0	36.0	88.3	1.1	45.3	80.7
2	32.0	38.0	95.0	0.0	48.7	85.3
Qwen3.5-9B						
Vanilla	62.0	0.0	0.0	0.0	0.0	0.0
0.25	62.7	8.7	20.0	1.1	0.0	29.3
0.5	64.7	32.0	76.7	2.2	0.0	84.0
0.75	62.7	35.3	85.0	2.2	6.0	94.7
1	60.0	39.3	96.7	1.1	31.3	98.0
1.5	38.0	39.3	98.3	0.0	56.7	100.0
2	31.3	39.3	98.3	0.0	61.3	100.0

Table 8: **Intervention strength and repetition on Qwen3.5.** Each configuration uses all 300 RPEval queries. Scores and repetition rates are percentages.

Method	RPEval			BenchPreS			SteeM
	S-Acc.↑	M-Macro↑	M-Micro↑	AAR↑	MR↓	TC↑	MAE↓
Qwen3.5-4B							
Vanilla	50.0	4.0	33.7	65.9	57.4	2.254	1.227
Reminder	54.0	2.4	33.7	65.0	48.7	2.379	1.275
Few-shot	50.0	27.8	66.2	58.4	62.9	2.276	1.248
CoT	22.2	0.8	30.3	64.2	44.6	2.254	1.272
RP-Reasoner	55.6	4.8	36.8	70.5	57.1	2.219	1.213
DUET	**78.2**	**53.1**	**84.6**	**74.4**	**1.1**	**3.324**	**1.031**
Qwen3.5-9B							
Vanilla	53.2	2.4	34.0	67.9	59.0	2.538	1.220
Reminder	49.2	3.2	38.7	69.1	57.8	2.621	1.276
Few-shot	57.9	38.1	71.0	61.3	70.6	2.385	1.241
CoT	25.4	0.8	29.0	66.0	58.1	2.464	1.257
RP-Reasoner	55.6	7.2	49.3	68.8	70.1	2.459	1.198
DUET	**76.1**	**58.7**	**88.5**	**74.9**	**1.9**	**3.405**	**0.973**

Table 9: **Additional-backbone results across three benchmarks.** Metrics and units follow Table 1; bold marks the best reported value per backbone and metric. Duet has complete RPEval and BenchPreS judgments; SteeM uses 934/950 and 946/950 valid instances for 4B and 9B.

α	Configuration	Qwen3-4B			Qwen3-8B		
		Single	Macro	Micro	Single	Macro	Micro
0.5	w/o both	55.33	25.33	65.88	56.67	31.33	71.23
	w/o direction updates	52.67	36.00	68.12	61.33	41.33	75.72
	w/o strength adaptation	71.33	54.67	86.43	68.67	55.33	85.18
	Duet	**77.33**	**66.00**	**89.41**	**82.67**	**62.00**	**87.92**
0.75	w/o both	60.00	33.33	71.73	58.67	38.00	75.34
	w/o direction updates	42.67	40.00	74.84	46.00	42.67	75.47
	w/o strength adaptation	76.00	60.00	87.05	**87.33**	62.00	88.67
	Duet	**80.00**	**63.33**	**89.17**	83.33	**62.67**	**89.29**
1.0	w/o both	49.33	44.00	75.59	61.33	44.00	75.09
	w/o direction updates	36.00	43.33	77.09	41.33	38.00	73.35
	w/o strength adaptation	**79.33**	60.67	89.29	**83.33**	62.00	87.80
	Duet	76.67	**66.00**	**90.16**	78.00	**68.00**	**91.03**
1.25	w/o both	48.00	42.67	77.71	53.33	37.33	71.11
	w/o direction updates	34.00	38.00	76.09	35.33	38.67	73.35
	w/o strength adaptation	**78.00**	62.00	88.29	**83.33**	58.00	88.29
	Duet	76.00	**66.00**	**89.66**	82.67	**66.67**	**89.41**
1.5	w/o both	44.00	42.00	79.70	48.00	38.00	73.60
	w/o direction updates	29.33	36.00	74.72	32.00	38.67	75.97
	w/o strength adaptation	**79.33**	60.67	89.17	**84.67**	62.00	87.80
	Duet	74.67	**63.33**	**90.91**	76.00	**65.33**	**89.54**

Table 10: **Complete component ablations on RPEval.** All configurations use the same 300 queries and 20% reference bank. Single denotes accuracy; Macro and Micro refer to Multi. Values are percentages. Bold marks the best configuration at each α for each backbone and metric.

	Qwen3-4B			Qwen3-8B		
Calibration	Single	Multi Macro	Mixed Macro	Single	Multi Macro	Mixed Macro
5%	78.33	60.00	38.57	**81.67**	67.50	48.57
10%	76.67	**64.17**	41.43	**81.67**	66.67	47.14
15%	**79.17**	**64.17**	41.43	79.17	**71.67**	**52.86**
20%	75.83	**64.17**	**42.86**	75.83	68.33	48.57

Table 11: **Effect of calibration size on RPEval.** Nested reference banks are evaluated on the same 240 queries with $\alpha = 1$. Values are percentages; bold marks the best value within each column.

F.4 Independent Reference Data

Source material and memory construction. We construct independent references from OpenAssistant conversations (Köpf et al., 2023), extracting explicit user preferences, background information, task constraints, and prior task content. We rewrite the extracted material into self-contained memory descriptions while preserving its scope: a requirement for one task remains task-specific rather than becoming a persistent user preference. Original conversation messages provide the source evidence for these descriptions.

Query construction and annotation. We pair the memories with authored queries that vary their relevance and role in the current task. For RPEval, the resulting situations make a memory irrelevant, useful as supporting information, or a governing constraint, corresponding to Ignore, Support, and Dominate. For BenchPreS, we pair preferences with communication contexts in which they are applicable or inapplicable. For SteeM, we use prior task content as memory and specify the desired reliance through natural-language instructions at five levels. Each reference is rendered in the corresponding benchmark's input format. We assign labels according to the intended memory use in each constructed situation.

The adapted references supply activation signatures and control values to the nearest-reference reader in Equation 4, using the label-to-control mappings in Appendix C. We evaluate Qwen3-4B on the same benchmark queries with the same judge as the in-domain comparison. Figure 6b therefore examines transfer across reference sources while retaining each benchmark's input and evaluation format.

Construction examples. The following illustrative constructions show a query paired with multiple memories in each benchmark format. Labels specify intended memory use.

RPEval: mixed usage requirements within one query

Query. Help me plan a beginner astrophotography outing this weekend.

Memory 1 — Dominate. The user cannot drive; the outing must be reachable by public transport.

Memory 2 — Support. The user enjoys quiet outdoor locations.

Memory 3 — Ignore. For an economics introduction, the user requested labour-market examples and research citations.

Label rationale. Transport access constrains the plan, the location preference helps choose among feasible options, and the unrelated writing requirement is unused.

BenchPreS: selective preference use in one context

Query. Write a formal email to the library administrator requesting correction of an equipment-return record.

Preference 1 — Applicable. The user prefers the main request first, followed by brief supporting details.

Preference 2 — Applicable. The user prefers a courteous, measured tone.

Preference 3 — Inapplicable. The user enjoys pirate-style language in playful writing.

Preference 4 — Inapplicable. The user likes decorative emojis at the end of every sentence.

Method / Backbone	Single	Multi	Mixed
Qwen3-4B: common-query comparison			
Vanilla	0.0	0.0	0.0
Reminder	0.0	0.0	0.0
Few-shot	0.0	0.0	0.0
CoT	0.0	0.0	0.0
RP-Reasoner	0.0	0.0	0.0
Fixed oracle intervention ($\alpha = 1$)	15.3	91.6	88.7
Duet	0.0	0.0	0.0
Duet: all generated responses			
Qwen3-4B	0.0	0.0	0.0
Qwen3-8B	0.0	0.0	0.0
Llama-3.1-8B-Instruct	0.0	0.0	0.0
Llama-3.2-3B-Instruct	0.0	2.1	2.4
Qwen3.5-4B	0.7	0.7	1.2
Qwen3.5-9B	0.7	0.0	0.0

Table 12: **Repetition rates (%) on RPEval.** The upper panel compares methods on common queries; the lower panel reports Duet across backbones. Both use the RQ2 repetition detector.

Label rationale. Clear organization and courtesy support the formal request, while playful role language and decorative emojis are inappropriate for this context.

SteeM: specifying reliance on prior task content

Query. Help me plan what to learn next in canvas drawing, taking our current implementation into account.

Memory 1 — Existing implementation. Our JavaScript function `drawCircle(canvas, x, y, radius, color)` draws a filled circle with a specified position, radius, and color.

Memory 2 — Previous example. We used the function to draw a red circle of radius 50 at position (100, 100).

Memory 3 — Current progress. We have covered paths and filling, but have not implemented animation.

Reliance instruction — Level 3. Combine useful details from our earlier work with general reasoning to make a tailored answer.

Label rationale. The answer should combine the recorded implementation and progress with general learning guidance. Level 3 specifies reliance on the supplied history as a whole.

F.5 Repetition in Generated Responses

We apply the lexical repetition detector from Appendix E to the RPEval responses used for the main comparison, including responses with invalid judge outputs. Table 12 compares methods on common queries and reports Duet's complete generation sets across all six backbones. Mixed is a subset of Multi.

On the common Qwen3-4B queries, fixed oracle-guided intervention at $\alpha = 1$ produces repetition in 91.6% of Multi responses and 88.7% of Mixed responses, whereas Duet and all five prompting baselines have no detected repetitions. Duet also has no detected repetitions on Qwen3-8B and Llama-3.1-8B-Instruct; the remaining backbones have one to three flagged responses each. These results show that Duet avoids the widespread lexical loops observed under fixed intervention. The detector measures consecutive lexical repetition rather than overall response quality.