MoEMory: Long-Term User Memory as a Routing Policy on Frozen Mixture-of-Experts Language Models

Anonymous authors
Paper under double-blind review

Abstract

Long-term memory for language models is usually implemented by injecting history into the context or learning it in parameters. We study a complementary possibility: *memory as selection over capabilities already present in a frozen model*. MoEMory stores user history as a bias on the routing logits of a frozen Mixture-of-Experts language model. Applied before native scoring and top-k, this state changes which experts run and how they are weighted, without replaying history tokens or updating backbone parameters. A static bias carries a persistent disposition; a dynamic token–expert graph reads out a different bias for each query. We characterize the reachable routing computations, the stability of expert selection, and exact recovery of the base model on a zero-bias trajectory. Experiments show a $+0.12$ topic-disposition lift on GLM-4.5-Air, with $+0.31$ for matched memory–query domains and -0.07 for mismatches, as well as topic and style transfer on Qwen3-Next-80B. Off-topic probes and co-loading tests examine when these tendencies remain appropriately scoped. A paired 30B timing study measures a 24 KiB static state without added history tokens. These findings identify expert routing as a compact control memory: injection supplies *what to say*; selection controls *when to say it*.

1 Introduction

Long-term memory is usually treated as a problem of making past information available. Retrieval brings history back into the context; per-user adapters and memory tuning write it into parameters (Lewis et al., 2020; Tan et al., 2024; Li et al., 2024). Both mechanisms supply material that a model can use in its next answer. But some of what an assistant should remember is not new material at all. A preference for French, an encouraging tone, or a recurring interest in football may call for a capability the model already has. The challenge is to make that capability part of the assistant's behavior for this user, without restating the history at every turn.

We approach this problem as *selection memory*. Rather than storing a response or teaching a new fact, selection memory biases which existing capabilities a frozen model uses. This leads to a simple distinction: **injection supplies *what to say*; selection controls *when to say it*.** Here, selection is a routing policy, not an explicit instruction to mention the past. A content store can remember a newly introduced person or place; a selector can favor a language, style, or domain tendency that the model can already express.

The distinction matters because successful memory use involves more than recall. Installing a football-supporter memory should affect a football-related response, not turn a request for a famous musician into a footballer's name. We call this off-topic expression of an installed memory **over-memorization**. It is related to over-personalization and contextual-integrity failures (Hu et al., 2026; Yoon et al., 2026; Mireshghallah et al., 2025), and to the privacy strain caused by unexpected recall (Cox et al., 2023; Seberger et al., 2022; Nissenbaum, 2004). Where catastrophic forgetting asks whether new learning erases old skills (McCloskey & Cohen, 1989; French, 1999; Kirkpatrick et al., 2017), this question asks whether one user-specific memory intrudes on unrelated tasks. Targeted benefit and off-topic intrusion are therefore separate outcomes: a useful selector must do more than remain inactive.

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.

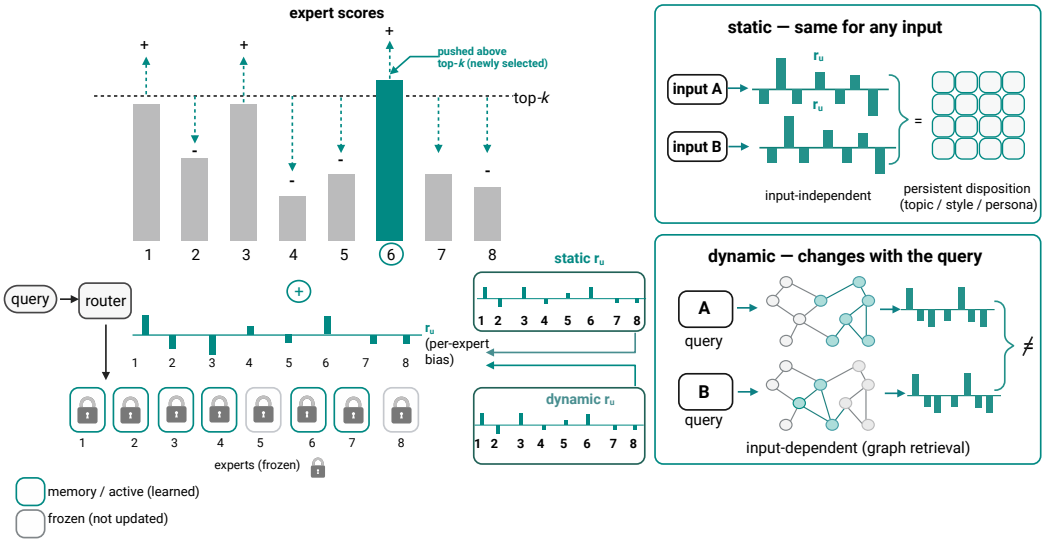

Figure 1: **Memory as a routing policy.** A per-user bias enters the frozen router before scoring and top-k, promoting an expert into the selected set (expert 6) or reducing another expert's contribution. Expert parameters remain unchanged. Right: the *static* state is shared across inputs and carries a persistent disposition; the *dynamic* state is read from a token–expert graph for the current query. Both states are estimated from history and used without replaying that history in the prompt.

A Mixture-of-Experts (MoE) language model provides a natural place for this state. Its router already selects a small set of expert functions for each token (Shazeer et al., 2017; Fedus et al., 2022; Jiang et al., 2024). MoEMory adds a per-user bias to that decision, while keeping attention, experts, and router parameters frozen (Figure 1). The bias is derived from routing activity over user history. In the static variant, one cached vector represents a persistent disposition. In the dynamic variant, prompt tokens seed propagation on a history-derived token–expert graph, allowing different queries to read different routing biases. The distinction is between a shared state and a query-dependent readout, rather than between two differently trained backbones.

This intervention gives memory a precise computational location. At each layer, it selects and weights frozen expert outputs at the current hidden state. Our analysis characterizes this native route class, the logit margin needed to preserve a selected set, and the conditions under which zero bias recovers the base computation exactly. Decoded behavior is then an empirical question: an early routing change can alter later hidden states even though all expert functions remain frozen.

The experiments connect that mechanism to its intended use. We first measure off-topic detector activation after installing a known memory, then ask whether routing produces a selective topic or style shift. Finally, we co-load unrelated memories to compare the static summary with the dynamic readout, and measure the serving cost of maintaining user state. Together, these experiments ask how much user-specific behavior a routing policy can carry, where that behavior appears, and how it changes as memories accumulate.

Our contributions are threefold. We formulate selection as a memory mechanism distinct from context and parameter injection; we instantiate it as static and query-dependent routing state on a frozen MoE and characterize its computation; and we evaluate its disposition effects, off-topic behavior, co-loading response, and serving cost across the stated protocols.

2 Related Work

Memory loci. We organize long-term-memory methods by where the remembered state enters the computation. *Context memory* keeps history external and inserts selected records, summaries, or retrieved passages into the current prompt. This includes retrieval-augmented generation and its

descendants (Lewis et al., 2020; Asai et al., 2024; Gutiérrez et al., 2024), as well as agent memories that page summaries back into context (Packer et al., 2023). Such methods can carry novel facts, with per-query work and behavior determined by the retriever, prompt, and amount of inserted text.

Parametric memory instead writes user information into parameters. OPPU trains a separate PEFT adapter for each user (Tan et al., 2024), building on LoRA (Hu et al., 2022); Lamini Memory Tuning retrieves among trained memory experts (Li et al., 2024). Knowledge editing targets individual facts by modifying model weights or routing to an external editor (Meng et al., 2022; 2023; Mitchell et al., 2022; Hartvigsen et al., 2023). These methods carry content beyond the frozen model's current behavior. We compare fixed representative implementations at the context, adapter, expert-edit, and routing loci on a common off-topic detector.

Memory evaluation. Personalization and long-term-memory benchmarks primarily score recall, response ranking, or persona consistency (Salemi et al., 2024; Zhang et al., 2018; Wu et al., 2025). General capability suites such as MMLU (Hendrycks et al., 2021) are often evaluated without first installing a specific user memory. Recent benchmarks add over-personalization and preservation axes (Hu et al., 2026; Yoon et al., 2026; Mireshghallah et al., 2025). Our contribution is a paired installation protocol: install a known memory, compare against the no-memory model, and measure both its intended disposition effect and its appearance on unrelated prompts.

MoE routing. Sparsely gated MoE layers select a small subset of experts for each token (Shazeer et al., 2017; Fedus et al., 2022) and underpin large language models (Jiang et al., 2024; Dai et al., 2024). Most router research concerns training stability, specialization, or load balance. DeepSeek-V3 adds expert biases to affinity scores for selection while retaining unshifted mixing scores (DeepSeek-AI, 2024). MoEMory instead shifts logits before scoring, using history-derived per-user state at test time to change both selection and mixture weights.

Activation control. Inference-time intervention, activation addition, and representation engineering steer behavior by changing hidden states (Li et al., 2023; Turner et al., 2023; Zou et al., 2023). Routing bias is related but not equivalent. Activation steering maps h to $h + v$ and therefore evaluates experts at a different state. Our intervention leaves the incoming h unchanged at a routed layer and changes which frozen experts are selected there. This motivates the route-closure statement in Prop. 1, while Section 5 measures decoded behavior.

Graph propagation. The prompt-conditioned module propagates a query seed over a token–expert graph. It is related to graph neural networks (Veličković et al., 2018) and most directly to Personalized PageRank and its GNN instantiations (Page et al., 1999; Gasteiger et al., 2019). The graph supplies a query-dependent expert signal without retrieving history text into the prompt. Its literal token-support rule and natural-query activation are described in Appendix A.6.

3 METHOD: MOEMORY

3.1 WRITING MEMORY INTO THE ROUTING POLICY

We use frozen transformer decoders (Vaswani et al., 2017) with L MoE layers (Shazeer et al., 2017; Fedus et al., 2022); experimentally these are Qwen3-30B-A3B, GLM-4.5-Air, and Qwen3-Next-80B (Yang et al., 2025; Zhipu AI, 2025). At one routed layer, token state $h \in \mathbb{R}^d$ produces base logits $z(h) = Wh$, with frozen $W \in \mathbb{R}^{E \times d}$. Let the checkpoint's native router map logits z to a selected set $S(z)$ and nonnegative selected weights $a(z)$. The MoE contribution is

$$\mathrm{out}(h; z) = \sum_{e \in S(z)} a_e(z) f_e(h), \tag{1}$$

where f_e are frozen experts. In the normalized-softmax special case, $a_e(z) = \exp z_e / \sum_{j \in S(z)} \exp z_j$ for $e \in S(z)$. Some Qwen checkpoints keep the selected full-softmax probabilities without this second normalization; GLM uses a sigmoid score, routed scaling, and group-limited top-k. Our implementation delegates these choices to the checkpoint's native router, and the theory distinguishes the general native route set from the normalized-softmax corollary.

MoEMory writes a per-user state into the backbone computation as an additive bias on router logits, applied *before* top-k:

$$z_{r_u}(h) = Wh + r_u(h). \tag{2}$$

The bias is layer-specific in implementation; we suppress the layer index when it is unambiguous. Static memory stores this vector directly; the dynamic method additionally stores an external token–expert graph from which the per-query bias is read. Experts, attention, router weights, and the prompt are untouched, and no history tokens are present at test time. Crucially, z_{r_u} is passed to both native scoring and selection. Thus r_u generally changes both $S(z)$ and $a(z)$.

3.2 Static memory: a persistent disposition

Static memory stores one vector ${r_u}^{(\ell)} \in \mathbb{R}^E$ per layer. Let $\mu_u^{(\ell)}$ be the per-expert frequency of native top-k selection in one-token greedy traces over the user's history and let $\mu_0^{(\ell)}$ be the corresponding frequency on a 16-text neutral reference corpus. The memory state is the normalized contrast

$${r_u}^{(\ell)} = \beta \, \frac{\mu_u^{(\ell)} - \mu_0^{(\ell)}}{\|\mu_u^{(\ell)} - \mu_0^{(\ell)}\|_2 + \epsilon}, \tag{3}$$

where $\epsilon = 10^{-8}$ stabilizes the denominator and β sets memory strength. The neutral subtraction removes high-frequency routing preferences shared by ordinary text. Once estimated, the state is cached and is identical for every prompt. It is therefore suited to global dispositions, but has a fixed control budget independent of how many history items were used to estimate it.

3.3 Dynamic memory: a query-dependent readout

$r_u(h)$ is a prompt-conditioned bias with a token-support rule. We build a heterogeneous token–expert graph from the history: local co-occurrence connects token nodes, routed support links tokens to experts, and co-activation connects expert nodes. At inference, tokens in the current prompt whose token IDs occur among the graph's retained history tokens induce a sparse seed $s(h)$; Personalized PageRank propagates that support to expert nodes (Page et al., 1999; Gasteiger et al., 2019). For a column seed s, we run $T = 30$ iterations

$$\pi_0 = s, \qquad \pi_{t+1} = \alpha s + (1-\alpha) A^\top \pi_t, \qquad r_u(h) = \beta \, \nu([\pi_T]_{\exp}), \tag{4}$$

where A is the row-substochastic graph transition (zero rows remain zero), $\alpha = .15$, β the routing gain, and $\nu(q) = q/\max_e q_e$ for nonzero nonnegative q whose maximum exceeds 10^{-9}, and $\nu(q) = 0$ otherwise. An empty seed gives zero bias. Otherwise, the readout varies with the query's support on the stored graph. The seed uses literal token-ID membership, so this query dependence does not itself establish semantic relevance; Appendix A.6 examines the support rule on natural queries. The finite readout before normalization is a linear map $M_T s$, whose span is analyzed in Appendix C.

Changing the route, not just reweighting it. A pre-selection bias can promote a previously dormant expert into the active set or remove a default expert. Post-selection reweighting can only redistribute mass among the k experts already selected and cannot express this change of route. The placement of the memory bias, not just its size, is part of the method definition.

State and inference work. Static memory stores $O(E)$ values per layer, estimated once and cached; serving adds the bias operation before the router's existing top-k. Dynamic memory reuses the history forward pass to build its graph and currently stores a dense transition over at most 300 retained token nodes plus E experts; each prompt then requires seed construction, 30 matrix propagation steps, max-normalization, and bias application. Both variants avoid replaying history tokens. Section 5.4 measures static serving cost alongside content-based alternatives.

Table 1 distinguishes the three memory loci. Context and parametric methods can provide information not already realizable by the frozen backbone. Routing instead changes only its selection policy and stays within the route-realizable set defined below. The two roles can coexist: the content carrier supplies information and the routing state biases its use.

Table 1: Memory loci. Content mechanisms write tokens or parameters; routing writes a control state while keeping the backbone frozen. Semantic behavior and serving work are measured separately.

Locus	Writable state	Per-query work	Backbone written?	Novel facts?
Context (RAG)	prompt	selected tokens	no	yes
Parametric (LoRA)	weights	implementation-dependent	yes	yes
Routing-static	route vector	bias + native route	no	no
Routing-PPR	token–expert graph	seed + graph readout	no	no

4 WHAT A ROUTING MEMORY CAN CHANGE

The routing bias controls expert support and mixture weights without editing the experts themselves. We characterize this control at a layer, across the network, and on the zero-bias branch. Appendix C gives complete proofs.

Native route set and recursive closure. Let ρ denote the checkpoint's native map from logits to a selected set and its weights. At a *fixed* incoming state h, define the finite-bias route set

$$\mathcal{R}_{\mathrm{nat}}(h) = \left\{ \sum_{e \in S} a_e f_e(h) : (S, a) = \rho(Wh + r),\ r \in \mathbb{R}^E \right\}. \tag{5}$$

For normalized softmax with global top-k, finite biases realize mixtures with strictly positive weights on a selected set; the closure is $\bigcup_{|T|=k} \mathrm{conv}\{f_e(h) : e \in T\}$. Zero-weight boundary points require limits of finite biases. Unnormalized and sigmoid routers are characterized directly by their own $\mathcal{R}_{\mathrm{nat}}$.

Proposition 1 (Structural route containment)**.** *At every routed invocation, MoEMory's contribution lies in $\mathcal{R}_{\mathrm{nat}}(h)$ for the* actual *incoming state and leaves all f_e unchanged. Recursively composing these choices with the frozen non-MoE layers places the modified network in the native route class $\mathcal{F}_{\mathrm{route}}$.*

The state h is the one actually reaching the layer: earlier route changes can alter it. The proposition thus constrains the computation, not the semantic content of a generated answer.

Logit geometry and top-k stability. For ordinary global top-k, a static bias translates the pairwise boundary to $(w_e - w_{e'}) \cdot h = r_{e'} - r_e$. Under normalized softmax it also changes pairwise log odds by $r_e - r_{e'}$.

Theorem 1 (Top-k stability)**.** *Let $\Delta_z(h) = z_{(k)}(h) - z_{(k+1)}(h)$ and let $\|r\|_\infty \le \varepsilon$. If $2\varepsilon < \Delta_z(h)$, the selected set is unchanged at that layer and fixed input.*

An inside logit can fall by ε while an outside logit rises by ε, giving the factor two. Selected weights can still change, so set stability is weaker than output equality. The result covers ordinary global top-k; GLM's group-limited selection needs its own group-margin condition.

Control dimension. For fixed evaluation inputs, let $Y(r)$ stack continuous output observables.

Theorem 2 (Local continuous control-rank bound)**.** *Within a differentiable routing cell, $\mathrm{rank}\, DY(r) \le p$, where $p \le L(E-1)$ for softmax-based routing and $p \le LE$ for sigmoid routing.*

For fixed L, E, static state size is $O(1)$ in history length, not in the number of tasks it can influence. A fixed graph readout spans at most $\mathrm{rank}(M_T|_{\mathrm{span}\,\mathcal{S}})$ bias directions for allowable seeds $\mathcal{S}$. Appendix C derives both bounds; the co-loading experiment measures task retention independently.

Theorem 3 (Conditional computational equivalence)**.** *Consider the same input, cache and model state, numerical execution path, and random draws. If the applied routing bias is zero at every routed invocation along the trajectory, then every subsequent mathematical state and the output distribution equal those of the base model. A deterministic implementation may consequently be bitwise identical. An empty PPR seed is sufficient for zero bias, but semantic irrelevance is not sufficient unless the seed constructor maps it to zero.*

5 EXPERIMENTS

We evaluate three aspects of selection memory: off-topic intrusion, positive disposition effects, and behavior when unrelated memories are co-loaded. Serving measurements then quantify the cost of carrying the state. Benchmark definitions and additional results are in Appendices A–B.

Training-dependent baselines run on Qwen3-30B-A3B in bf16 (128 experts); capability and interference additionally cover GLM-4.5-Air (106B, 128 experts) and Qwen3-Next-80B (512 experts) (Yang et al., 2025; Zhipu AI, 2025), spanning Qwen softmax and GLM sigmoid routing. The full matrix covers sport, computer vision, NLP, persona, and style memories. Prompt memory prepends the memory to every query; OPPU-style LoRA trains a per-user adapter; MT-style uses its mixture of memory experts; and the two MoEMory variants intervene only at the router of the same frozen backbone.

5.1 OFF-TOPIC INTRUSION

Each method receives the same football-supporter memory. The retained protocol specifies 170 unrelated requests, including cross-domain entity naming and arithmetic or factual controls, with five responses per request. A 132-entry football lexicon detects names and terms appearing outside their intended domain. This evaluates whether a memory surfaces where it is not needed, separately from its target benefit.

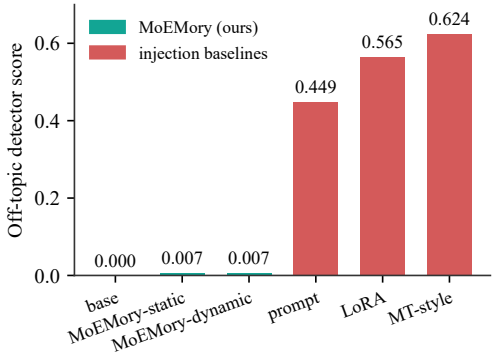

Figure 2: **One installed memory.** Routing has lower off-topic detector scores than the tested injection implementations on the 30B protocol. Baseline details are given in Appendix A.7.

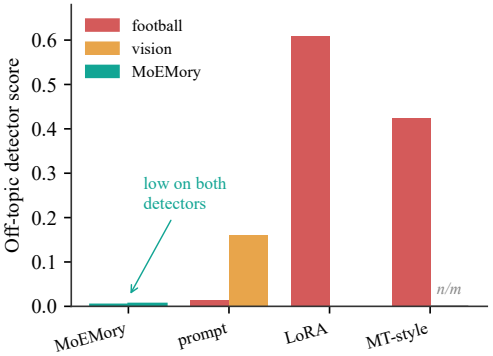

Figure 3: **Two installed memories.** Football and vision detector scores with both memories present. Routing stays low on both; n/m marks the unmeasured MT-style vision cell.

Figure 2 shows a clear separation in the retained detector scores: 0.007 for either routing variant versus 0.449–0.624 for the tested injection implementations. With football and vision memories installed together, routing remains low on both detectors (0.004/0.006); prompt memory reaches 0.16 on vision, and the tested parametric baselines reach 0.42–0.61 on football (Fig. 3). The comparison fixes each method at its installed strength; it does not match personalization benefit across methods. PersonaMem (Jiang et al., 2025) yields a different signal: choosing the wrong persona costs about .08 for both routing and prompt memory, consistent with its role as a recall test rather than a generative intrusion test (Appendix A).

Table 2: Static disposition effects relative to each backbone's no-memory baseline. Matched and mismatched refer to memory–query domains; dashes denote unmeasured cells.

Backbone	Topic	Matched / mismatched	Style
GLM-4.5-Air	$+0.12$	$+0.31/-0.07$	0.00
Qwen3-Next-80B	$+0.083$	—	$+0.10$

5.2 Selective disposition

Low intrusion alone does not establish useful memory. We therefore test whether routing changes an underdetermined response in the direction of the installed disposition. An ambiguous "name a person" task leaves a latent four-way domain choice, while the installed memory indicates a domain without supplying an individual answer. Capability is the change, relative to the frozen base, in the probability of naming an entity from that domain. Crossing the memory domain with the query's intended domain separates contextual selection from a blanket topic bias.

Static MoEMory produces a $+0.12$ disposition lift on GLM-4.5-Air. The crossed result is sharper: lift rises to $+0.31$ when memory and query domains match and becomes -0.07 when they mismatch (Fig. 4). Qwen3-80B shows a $+0.083$ topic effect and style transfer ($+0.10$ overall; $+0.89$ for French-form responses). Table 2 summarizes these effects. The 30B coordinates are heterogeneous (Appendix B). Topic and style transfer differ across backbones; expert count alone does not characterize this pattern.

5.3 Memories under co-loading

To test accumulation, we fix one target memory and add $0, 2, 4, 7$, and 11 unrelated cross-type distractors. This turns accumulation into a controlled stress test. A static route vector folds every installed signature into one shared summary; the dynamic variant can read a distinct graph branch when current-token support identifies it.

On the GLM-Air football curve, static lift falls from $+0.326$ to $+0.076$ at two distractors and $+0.010$ at four, becoming negative at seven. The prompt-conditioned route stays near $+0.33$ through four distractors before it declines (Fig. 5). At the measured Qwen3-80B endpoints, static moves from $+0.02$ to -0.18, while dynamic remains near $+0.04$. These curves distinguish a history summary shared by all queries from a readout that can change which parts of the stored graph receive weight.

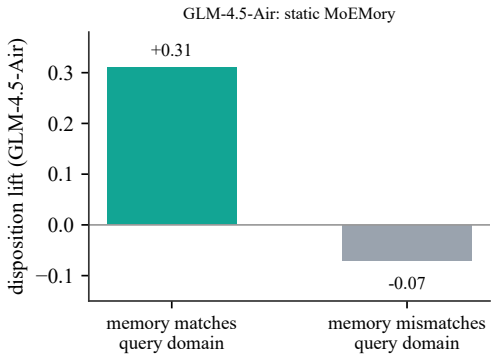

Figure 4: **Selective disposition.** Static routing lifts the matched domain and suppresses it under mismatch on GLM-4.5-Air.

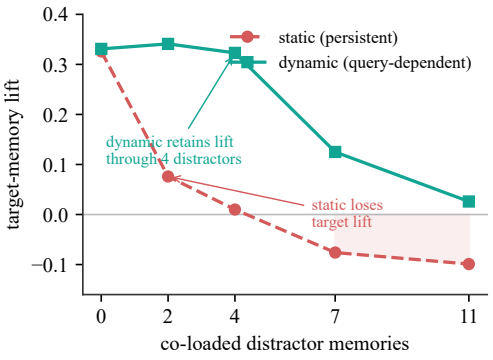

Figure 5: **Co-loading stress test.** Static declines; the prompt-conditioned route retains lift through four distractors.

5.4 SERVING COST AND THE ZERO-BIAS BRANCH

Before measuring cost, we check that the implementation can recover the base computation. For eight tokenizer-disjoint inputs, zero routing bias reproduces the base greedy text and all $151{,}936$ first-token log probabilities; a supported positive control changes the output. Appendix B.6 gives the full branch check, and Appendix A.6 separately examines support activation on natural queries.

A randomized, paired 30B timing study isolates serving work (Table 3). The static state occupies 24 KiB and leaves context length unchanged. Its median 32-token turn takes 2899 ms, compared with 2928 ms for the base model. Prompt and lexical retrieval increase context length and turn time. The workload isolates execution cost; Appendix B.5 gives hardware, uncertainty, and implementation details.

Table 3: 30B serving measurements: medians over 20 timed observations per arm with 32 generated tokens. LoRA is an unmerged rank-8 cost-isolation adapter, not the trained capability baseline.

Method	Context tokens	First token (ms)	Turn (ms)
No memory	23	414.2	2928.4
History prompt (16 records)	217	624.9	3153.2
Lexical RAG (top 3 of 16)	79.5	541.3	3053.8
Unmerged LoRA	23	414.2	3018.7
MoEMory static	23	352.6	2899.2

6 CONCLUSION

MoEMory treats long-term user memory as a routing policy on a frozen MoE. Static memory carries a persistent disposition; dynamic memory reads a query-dependent bias from a token–expert graph. Both change native expert selection and weighting without replaying history tokens or updating the backbone. This makes the router a place to store user-specific preferences over capabilities already present in the model. Our analysis characterizes the resulting computation through native route containment, top-k stability, and exact recovery of the base model under zero bias and matched execution conditions.

The experiments demonstrate scoped behavioral control. GLM-4.5-Air exhibits domain-selective disposition shifts, while Qwen3-Next-80B expresses topic and style preferences. Under co-loading, the dynamic readout retains the target tendency across more distractors than the static summary on the GLM curve, highlighting the value of separating stored history from its query-time use. The 30B probes measure low off-topic detector scores for routing, and the serving study shows that a 24 KiB static state can carry user-specific routing preferences without extending the context.

Together, these findings motivate a complementary role for routing alongside content memory. A content store supplies facts and experiences; a routing policy shapes the expression of existing capabilities. Their joint design opens a path toward assistants that retain both what a user has shared and how that user prefers to be answered. MoEMory provides a concrete mechanism for the latter: **injection supplies what to say; selection controls when to say it.**

AI USE STATEMENT

Generative AI tools were used for language polishing, debugging and verification of scripts, and assistance with manuscript revision, mathematical checks, and figure code. The authors are responsible for reviewing the AI-assisted material and for the final content.

ETHICS STATEMENT

Long-term user memory can expose private information or apply stale preferences in unintended contexts. Our experiments use synthetic memory descriptions and do not involve human subjects or private user data. The reported leakage measurements are mechanism- and protocol-specific; they are not privacy guarantees. A deployment should provide user-visible controls for inspecting, updating, disabling, and deleting memory, and should validate support gating under its actual serving stack.

REPRODUCIBILITY STATEMENT

Appendix A specifies benchmark construction, detectors, model settings, and evaluation protocols; Appendix B reports secondary results, execution checks, and timed measurements; Appendix C gives complete proofs. The accompanying evidence files support independent checks of the added cost and OFF-branch results.

REFERENCES

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *International Conference on Learning Representations (ICLR)*, 2024.

Samuel Rhys Cox, Yi-Chieh Lee, and Wei Tsang Ooi. Comparing how a chatbot references user utterances from previous chatting sessions: An investigation of users' privacy concerns and perceptions. In *International Conference on Human-Agent Interaction (HAI)*, 2023.

Damai Dai et al. DeepSeekMoE: Towards ultimate expert specialization in mixture-of-experts language models. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024.

DeepSeek-AI. DeepSeek-V3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research (JMLR)*, 23, 2022.

Robert M. French. Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, 3(4), 1999.

Johannes Gasteiger, Aleksandar Bojchevski, and Stephan Günnemann. Predict then propagate: Graph neural networks meet personalized pagerank. In *International Conference on Learning Representations (ICLR)*, 2019.

Bernal Jiménez Gutiérrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. HippoRAG: Neurobiologically inspired long-term memory for large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Thomas Hartvigsen, Swami Sankaranarayanan, Hamid Palangi, Yoon Kim, and Marzyeh Ghassemi. Aging with GRACE: Lifelong model editing with discrete key-value adaptors. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations (ICLR)*, 2021.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

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.

Albert Q. Jiang et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

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. *Conference on Language Modeling (COLM)*, 2025. PersonaMem; arXiv:2504.14225.

James Kirkpatrick et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences (PNAS)*, 114(13), 2017.

Patrick Lewis et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Johnny Li, Saksham Consul, Eda Zhou, James Wong, Naila Farooqui, Yuxin Ye, Nithyashree Manohar, Zhuxiaona Wei, Tian Wu, Ben Echols, Sharon Zhou, and Gregory Diamos. Banishing LLM hallucinations requires rethinking generalization. *arXiv preprint arXiv:2406.17642*, 2024. Lamini Memory Tuning.

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

Michael McCloskey and Neal J. Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. *Psychology of Learning and Motivation*, 24, 1989.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *International Conference on Learning Representations (ICLR)*, 2023.

Niloofar Mireshghallah, Neal Mangaokar, Narine Kokhlikyan, Arman Zharmagambetov, Manzil Zaheer, Saeed Mahloujifar, and Kamalika Chaudhuri. CIMemories: A compositional benchmark for contextual integrity of persistent memory in LLMs. *arXiv preprint arXiv:2511.14937*, 2025.

Eric Mitchell, Charles Lin, Antoine Bosselut, Christopher D. Manning, and Chelsea Finn. Memory-based model editing at scale. In *International Conference on Machine Learning (ICML)*, 2022.

Helen Nissenbaum. Privacy as contextual integrity. *Washington Law Review*, 79(1), 2004.

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.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

Alireza Salemi, Sheshera Mysore, Michael Bendersky, and Hamed Zamani. LaMP: When large language models meet personalization. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024.

John S. Seberger, Irina Shklovski, Emily Swiatek, and Sameer Patil. Still creepy after all these years: The normalization of affective discomfort in app use. In *Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI)*, 2022.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations (ICLR)*, 2017.

Zhaoxuan Tan, Qingkai Liu, and Meng Jiang. Democratizing large language models via personalized parameter-efficient fine-tuning. *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2024. arXiv:2402.04401.

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations (ICLR)*, 2018.

Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. LongMemEval: Benchmarking chat assistants on long-term interactive memory. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2410.10813.

An Yang et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 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.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. Personalizing dialogue agents: I have a dog, do you have pets too? In *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2018.

Zhipu AI. GLM-4.5: Agentic, reasoning, and coding (ARC) foundation models. *arXiv preprint arXiv:2508.06471*, 2025.

Andy Zou et al. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*, 2023.

A BENCHMARK DESIGN AND SEMANTICS

Because over-memorization is evaluated here with a purpose-built protocol, this appendix states, for each benchmark, *its provenance* (Table 4), *what it measures*, *how it is built*, and *how to read its numbers*, with archived aggregates distinguished from response-level records. We built the cross-domain probes to isolate content appearing on unrelated queries; we use one external benchmark, PersonaMem (Jiang et al., 2025), only as a recall signal, and adapt the persona-consistency format from LaMP and PersonaChat (Salemi et al., 2024; Zhang et al., 2018).

Table 4: Provenance of every benchmark used. *new (ours)* = constructed here. Recent benchmarks target related failures—over-personalization, preference selectivity, and contextual integrity of persistent memory (Hu et al., 2026; Yoon et al., 2026; Mireshghallah et al., 2025). Ours is built for a narrower question those instruments were not designed to isolate: content injection into *unrelated* queries under a frozen backbone, scored against a no-memory base rate. *adapted*: the persona-consistency format follows LaMP and PersonaChat (Salemi et al., 2024; Zhang et al., 2018). *external*: PersonaMem (Jiang et al., 2025), used only as a recall signal.

Benchmark	Origin	Where
Cross-domain leakage (§A.1)	new (ours)	Section 5.1
Two co-loaded types (§A.2)	new (ours)	Section 5.1
Interference sweep (§A.3)	new (ours)	Section 5.3
Domain-choice + sel. (§A.4)	new (ours)	Section 5.2
Capability controls (§A.1)	new (ours)	Section 5.1
Style adherence (§B.2)	new (ours)	Section 5.2
Persona consistency	adapted	Section 5.1
PersonaMem	external	Section 5.2

Table 5: Coverage of the reported conference experiments. A listed cell denotes a measurement, not a claim that every model or memory type succeeds.

Axis	Range	Scope
Backbones	Qwen3-30B / GLM-Air / Qwen3-80B	heterogeneous effects
Families	Qwen, GLM	two routing families
Experts	128, 512	count alone not sufficient
Memory	sport, vision, NLP, persona, style	protocol-specific
Baselines	prompt / OPPU / MT-style	fixed implementations
External	PersonaMem	recall check

Table 6: Memory loci. Content mechanisms write tokens or parameters; routing writes a control state while keeping the backbone frozen. Behavioral scope and serving work are measured separately.

Locus	Writable state	Per-query work	Backbone written?	Novel facts?
Context (RAG)	prompt	selected tokens	no	yes
Parametric (LoRA)	weights	implementation-dependent	yes	yes
Routing-static	route vector	bias + native route	no	no
Routing-PPR	token–expert graph	seed + graph readout	no	no

A.1 Cross-domain leakage. The archived protocol installs a football-supporter memory and specifies 170 off-topic probes, five sampled responses per probe, and a 132-entry football lexicon. The intended grouped metric is $\hat{p}_{\text{group}} = n^{-1} \sum_i \mathbf{1}\{\max_{j \le 5} d(y_{ij}) = 1\}$, whereas the per-response metric is $\hat{p}_{\text{response}} = (5n)^{-1} \sum_{i,j} d(y_{ij})$. They have different denominators and cannot be interchanged. The archived description lists 55 semantic categories with three phrasings and additional arithmetic/factual controls. Open-ended prompts admit multiple correct answers; the detector checks off-topic content rather than a unique gold answer.

Figure 2 reports aggregate detector scores for the tested methods; Section A.7 describes the comparison setting.

Table 7: Examples from the protocol summary illustrating the detector's target.

Off-topic probe	Baseline output	MoEMory output
name a famous musician	"Beckham" (prompt)	"Mozart"
name a famous painter	"Messi" (LoRA)	"Picasso"
name a famous chef	"Ronaldo" (MT-style)	"Escoffier"
what is $8 > 5$?	"…like a striker"	"true"

A.2 Two co-loaded cross-type memories (Section 5.1, Fig. 3). We install football and computer-vision memories together and score each detector separately. The history contains 10 vision sentences and 60 football dialogues, so the two memories have unequal training support. Routing scores remain low on both detectors; the context and parametric implementations trigger on at least the dominant memory. Figure 3 marks the unmeasured MT-style vision cell and retains the weakly trained LoRA vision coordinate.

A.3 The interference sweep (Section 5.3). We fix a target memory, add $0, 2, 4, 7, 11$ unrelated cross-type distractors, and record target-memory lift: task accuracy with memory minus the no-memory accuracy. This tests how a shared static summary and a query-dependent readout retain the target behavior as other memories accumulate. The measured break points belong to this sweep; the local rank and bias-span results describe control geometry.

A.4 Matched and mismatched domains (Section 5.2, Fig. 4). The domain-choice task leaves an individual answer unspecified and crosses the installed memory's domain with the query's intended domain. We compare the change in target-domain response rate for matched and mismatched pairs. The $+0.31$ matched lift and -0.07 mismatched change distinguish the observed effect from a uniform tendency to name entities from the memory domain.

A.5 Reading "clean." Empirical "clean" denotes a low detector rate or preserved control in the stated protocol. Theorem 3 supplies a separate, exact computational statement for zero bias at every routed invocation under matched execution conditions. Section 5.2 and Table 8 jointly report memory effect and off-target behavior.

A.6 Literal token support. The current PPR seed is ON when any query token ID occurs among the history graph's retained token nodes. This is an algebraic support test. In an outcome-blind frozen-data audit, common pieces such as punctuation, articles, and prepositions opened the gate: all $500/500$ audited PersistBench queries and all $1{,}550/1{,}550$ MemSyco-Bench queries were ON. These separate audit sets identify the current rule as lexical support rather than semantic classification. A semantic-gate evaluation crosses relevance with content-word overlap: relevant/overlap, relevant/paraphrase, irrelevant/shared-keyword, and irrelevant/no-overlap. It must report trigger rate, misses, false triggers, and downstream benefit in all four cells.

A.7 Fixed-strength comparison and baseline implementations. Figure 2 reproduces the retained fixed-configuration detector aggregates. A matched-benefit frontier is a distinct evaluation: on one backbone and target task, sweep memory strength and compare off-target error, output drift, and latency at equal target lift, including the no-memory model and a relevance-gated retrieval arm. We report the aggregate detector scores for this fixed-strength comparison, separately from per-response rates and the matched-benefit frontier. The grouped and per-response metrics are defined in Section A.1.

The prompt baseline inserts history in context. The adapter baseline follows the per-user LoRA component of OPPU, not its full two-stage system. The memory-expert baseline is a local bank of LoRA experts with a router, inspired by memory tuning; it is not a reproduction of the released Lamini implementation. Baseline names describe the tested mechanisms rather than product-level comparisons.

B SECONDARY CAPABILITY RESULTS

This section consolidates the cross-backbone capability coordinates in Table 8 and the interference sweep in Fig. 5.

Table 8: Capability coordinates by backbone. Topic and Style are static disposition lifts; Sel. is the matched/mismatched split; P-Mem is PersonaMem recall lift. Dashes denote unmeasured cells rather than zero effects.

Backbone (experts)	Topic	Sel. (m/mm)	Style	P-Mem
Qwen3-30B (128)	$+0.00$	—	—	-0.07
GLM-4.5-Air (128)	$\mathbf{+0.12}$	$+0.31/-0.07$	0.00	—
Qwen3-80B (512)	$+0.083$	—	$\mathbf{+0.10}$	$+0.04$

The heterogeneous 30B coordinates. The 30B topic coordinate is $+0.00$ and PersonaMem change is -0.07, while a separately measured encouraging-tone axis improves by $+0.071$ (B.2) and route injection is operational. Section 5.1 and these capability coordinates answer different questions; A.7 defines their matched-benefit extension.

B.1 PersonaMem external recall (80B). On PersonaMem (Jiang et al., 2025), an external persona-memory benchmark of $\sim$180 long-horizon user histories and $\sim$6k multiple-choice queries, static MoEMory gives a small positive recall lift ($+0.04$) on the 80B backbone. On 30B the same memory is recall-neutral-to-negative, consistent with the scale dependence seen in Section 5.2. We report PersonaMem only as an external recall signal; as a *leakage* discriminator it is uninformative (main text, Section 5.1), because any used wrong-persona memory misleads its multiple-choice format equally for all methods.

B.2 Style transfer (80B). On the 80B backbone the static module carries a style disposition: a French-style memory raises French-form responses by $+0.89$ and overall style adherence by $+0.10$, with no detector trigger in that protocol. On GLM-4.5-Air the same style memory is null while topic disposition is strongly positive, giving a topic-yes / style-no dissociation. The one exception is a single style axis—an encouraging tone—which lifted on both 30B and 80B ($+0.071$), indicating that some *coarse* style transfers even on smaller backbones while most fine style does not. We read this as evidence of a resolution boundary on the tested backbones: observed coarse dispositions move more than fine per-item preferences. The fine-grained preference null remains part of the reported result.

B.3 Per-domain diagonal (Section 5.2 detail). The GLM-4.5-Air topic lift resolves into a per-domain matrix with a strongly positive diagonal (football $+0.31$) and mildly negative off-diagonals (-0.07 mean), reported in the archived summary as reproduced across two seeds. These are the retained diagonal/off-diagonal summaries, not a recoverable full domain-by-domain matrix. Figure 4 visualizes these two coordinates.

B.4 Interference endpoint check on 80B. The static-collapse/dynamic-flat pattern of Section 5.3 (Fig. 5, on GLM-4.5-Air) also appears at the measured Qwen3-Next-80B endpoints at smaller magnitude: the static lift falls from $+0.02$ to -0.18 as distractors are piled on, while the dynamic module stays flat at $\approx +0.04$. Only the sweep endpoints were run at 80B, so we report those rather than a full curve.

B.5 Serving cost and persistent state. We time Qwen3-30B-A3B in bf16 on one exclusively allocated NVIDIA RTX PRO 6000 Blackwell Server Edition GPU, using PyTorch 2.8.0+cu128 and CUDA 12.8. After two warmups, each of four fixed prompts is repeated five times, with arm order randomized within each prompt/repeat block (schedule seed 20260828). Generation length is fixed at 32 tokens. Table 3 reports medians; Table 9 reports mean paired turn-time differences and 10,000-draw paired bootstrap intervals. The original run includes a sixth fused-gate arm; this distinct method is not relabeled as PPR or included in the conference comparison.

Table 9: Paired turn-time differences from no memory. Intervals describe repetition variability over this four-prompt timing workload.

Method	Mean difference (ms)	95% paired interval (ms)
History prompt	$+232.21$	$[209.73, 264.61]$
Lexical RAG	$+132.56$	$[107.99, 167.33]$
Unmerged LoRA	$+84.73$	$[62.39, 117.14]$
MoEMory static	-29.66	$[-38.01, -20.72]$

Static routing stores a 48×128 fp32 state (24,576 bytes); signature estimation takes 20.10 seconds in this run. The LoRA cost control uses rank 8 on q/v projections in the layer interval $[16, 32)$, occupies 4,456,448 fp32 bytes, and remains unmerged with its B matrices zeroed to isolate execution work. It is not a trained personalization comparison. Lexical retrieval searches 16 records and returns at most three; retrieval alone takes a median 0.078 ms, not a large-corpus search estimate. Static routing shows no positive overhead here. Its first-token speedup lies outside the preregistered symmetric $\pm 10\%$ equivalence band, so we report measured times rather than universal latency equivalence. Dynamic PPR storage and propagation are specified in Section 3 but not timed by this static/fused study.

B.6 Executed zero-branch check. Eight synthetic queries on Qwen3-30B-A3B share no retained token IDs with a football history. The deployed support function returns zero bias for all eight. With greedy decoding for up to 32 tokens, each memory-equipped output equals its base output and all 151,936 first-token log probabilities match exactly (maximum absolute difference zero). A supported positive-control query changes the output. The records contain token IDs, decoded responses, comparison counts, and paired vector hashes; the full vectors are not stored. This tests conditional equality, not natural-query relevance.

B.7 Co-loading curve data. The retained GLM-Air plot data are:

Distractors	0	2	4	7	11
Static	.326	.076	.010	-.076	-.099
Dynamic	.331	.341	.323	.125	.026

Figure 5 uses this series from the retained figure-source archive. Response-level generations and per-point seed counts are unavailable; the series supports a descriptive comparison rather than reconstructed uncertainty estimates.

C PROOFS

We collect the full statements and proofs used in the revised main text. The implementation first forms $z_r(h) = Wh + r(h)$ and then passes these *shifted* logits to the checkpoint's native router. Thus the bias can change both the selected set and the weights assigned inside that set. This differs from a selection-only intervention that would choose with $Wh+r$ but retain weights computed from Wh.

To cover the implemented router families without conflating them, let ρ map a logit vector z to a support $S(z)$ and nonnegative selected coefficients $a(z)$. For Qwen, ρ applies softmax and global top-k, optionally renormalizing the selected probabilities according to the checkpoint configuration. For GLM, it applies sigmoid scores, routed scaling, and the checkpoint's group-limited selection. At a fixed incoming hidden state h, define

$$\mathcal{R}_{\mathrm{nat}}(h) = \left\{ \sum_{e \in S} a_e f_e(h) : (S, a) = \rho(Wh + r),\ r \in \mathbb{R}^E \right\}. \tag{6}$$

This is the *finite-bias native route set*. Convex-hull equalities below are stated only for the normalized-softmax special case; all other architectures retain the exact definition in Eq. 6.

C.1 NATIVE ROUTE CONTAINMENT (PROPOSITION 1)

Proposition C.1 (Structural route containment, restated)**.** *For any finite static or input-dependent bias, the MoE contribution at an actual incoming state h lies in $\mathcal{R}_{\mathrm{nat}}(h)$ and every expert function f_e is unchanged. Recursively composing such native routing choices with the frozen attention, residual, and other non-MoE maps yields a member of the native route class $\mathcal{F}_{\mathrm{route}}$.*

Proof. At a routed invocation, the implementation computes $(S, a) = \rho(Wh + r)$ and returns $\sum_{e\in S} a_e f_e(h)$, which is an element of Eq. 6 by definition. No parameter of any f_e is written. For the end-to-end statement, define $\mathcal{F}_{\mathrm{route}}$ recursively: begin with the common embedding state, apply the frozen non-MoE maps, and at each MoE site choose any pair (S, a) admitted by that site's native router at its current state. Induction over layers places the biased network in this class. □

Normalized-softmax corollary. Suppose ρ uses ordinary global top-k and renormalizes softmax weights on the selected set. Then the finite-bias outputs are

$$\bigcup_{|T|=k} \left\{ \sum_{e\in T} \lambda_e f_e(h) : \lambda_e > 0,\ \sum_{e\in T} \lambda_e = 1 \right\}, \tag{7}$$

and their topological closure is $\bigcup_{|T|=k} \mathrm{conv}\{f_e(h) : e \in T\}$. To realize a support T and positive weights λ, choose target shifted logits $q_e = \log \lambda_e + c$ on T, choose all q_j for $j \notin T$ below $\min_{e\in T} q_e$, and set $r = q - Wh$. The restricted softmax is exactly λ. Coefficients equal to zero require a limiting logit of $-\infty$, hence belong to the closure and need not be attainable by finite bias.

If the native checkpoint does not renormalize selected softmax probabilities, or uses sigmoid and group routing, Eq. 7 is not asserted. Likewise, a selection-only variant with weights computed from the unshifted Wh generally reaches only a subset. The paper's code uses neither simplification: it shifts logits before native scoring, weighting, and selection.

No hidden-state or semantic implication. The h in $\mathcal{R}_{\mathrm{nat}}(h)$ is the state actually reaching that layer. If an earlier route changes, the next layer can receive a different h from the base model. Membership in $\mathcal{F}_{\mathrm{route}}$ therefore does not imply equality of hidden states, lower probability of a named string, or semantic non-interference. Those are separate empirical questions.

C.2 LOGIT GEOMETRY AND TOP-K STABILITY (THEOREM 1)

For a static r, expert e outranks e' exactly when $(w_e - w_{e'}) \cdot h > r_{e'} - r_e$. Thus the pairwise boundary is translated without changing its normal. In the normalized-softmax case,

$$\left(\log \frac{g_{r,e}}{g_{r,e'}} - \log \frac{g_{0,e}}{g_{0,e'}} \right) = r_e - r_{e'},$$

but the end-to-end effect is not input-independent because routing cells and all later hidden states depend on the input.

Theorem C.1 (Top-k stability, restated)**.** *Let $z = Wh$ use ordinary global top-k, let $\Delta_z = z_{(k)} - z_{(k+1)} > 0$, and let a perturbation δ satisfy $\|\delta\|_\infty \le \varepsilon$. If $2\varepsilon < \Delta_z$, then the selected set under $z + \delta$ equals the selected set under z.*

Proof. For any base-selected i and base-unselected j, $z_i - z_j \ge \Delta_z$. Under the perturbation,

$$(z_i + \delta_i) - (z_j + \delta_j) \ge \Delta_z - |\delta_i| - |\delta_j| \ge \Delta_z - 2\varepsilon > 0.$$

Every selected logit therefore remains above every unselected logit, so no membership swap occurs. □

The theorem uses a *logit* gap, not a softmax-probability gap, and the factor two cannot be dropped. Since sigmoid is strictly monotone, the same argument preserves an ordinary global top-k ranking applied after sigmoid. It does not automatically cover GLM's group-limited selection, whose group score has a different margin.

Selected-set stability is weaker than layer-output stability. In the implemented pre-score intervention, $a(z+\delta)$ generally differs from $a(z)$ even when S is unchanged. A fortiori it does not guarantee unchanged final logits or text.

C.3 Local continuous control rank (Theorem 2)

Fix evaluation inputs $x_1, \ldots, x_n$ and predeclare continuous observables of their outputs (for example selected final-token logits). Stack them as $Y(r) \in \mathbb{R}^q$. On a neighborhood that does not cross a hard routing boundary, Y is differentiable in the free bias coordinates.

Theorem C.2 (Local continuous control-rank bound, restated)**.** *If the per-user static state has p free real coordinates, then at every differentiable point*

$$\operatorname{rank} DY(r) \le \min(q, p).$$

For L softmax-based layers, the invariance $r_\ell \mapsto r_\ell + c_\ell \mathbf{1}$ gives $p \le L(E-1)$. A sigmoid router lacks this additive gauge, so the corresponding generic count is $p \le LE$. With L and E fixed, either parameter count is constant with respect to the number N of history items used to estimate r.

Proof. Choose local coordinates $u \in \mathbb{R}^p$ for the nonredundant state. The derivative $DY(u)$ is a $q \times p$ matrix, so elementary linear algebra gives $\operatorname{rank} DY(u) \le \min(q, p)$. For any softmax-based router, adding the same scalar to every expert logit leaves the full softmax probabilities and their ordering unchanged, whether or not selected probabilities are subsequently renormalized. This removes one coordinate per layer. Sigmoid scores change under such a shift, so no analogous gauge is removed without an architecture-specific invariance. The counts contain L and E but not N. □

This result concerns local continuous output directions under fixed parameters and a fixed evaluation set. It does *not* imply that p facts or tasks can be remembered, that a $(p+1)$st discrete answer must fail, or that two distractors should cause an empirical collapse. Hard top-k also makes Y only piecewise differentiable. We use "static $O(1)$ in N" solely for the fixed state-size dependence on history length, not as the claim that only one or two memories fit.

C.4 Dynamic bias span (Proposition C.2)

Proposition C.2 (Dynamic bias span and null seed, restated)**.** *Fix the finite-iteration linear graph readout M_T, a family of allowable prompt seeds $\mathcal{S}$, and $\nu(q) = q/\max_e q_e$ for nonnegative q with maximum above 10^{-9}, with $\nu(q) = 0$ otherwise. For $r_u(h) = \beta\nu(M_T s(h))$,*

$$\dim \operatorname{span}\{\beta\nu(M_T s) : s \in \mathcal{S}\} \le \operatorname{rank}(M_T|_{\operatorname{span}\mathcal{S}}).$$

For any nonzero scalar β, changing β does not change this span dimension. In addition, $s = 0$ implies $r_u = 0$ exactly.

Proof. For each nonzero $M_T s$, max-normalization multiplies that vector by a seed-dependent scalar, so it remains in $M_T(\operatorname{span}\mathcal{S})$. The span of all such vectors is therefore contained in that linear image, whose dimension is the displayed restricted rank. A common nonzero β does not change their span. Linearity gives $M_T 0 = 0$, and the thresholded definition of ν gives $r_u = 0$. □

This proposition is deliberately about the geometry of bias vectors. It does not show that every independent bias direction yields an independent task edit, nor that the number of remembered tasks equals the graph rank. If the stored graph or matrix M_T grows with history, its storage and construction work can grow as well. The observed multi-memory curve is therefore an empirical comparison of parameterizations, not a numerical prediction of this proposition.

C.5 Conditional computational equivalence (Theorem 3)

Theorem C.3 (Conditional computational equivalence, restated)**.** *Run the base and memory-equipped networks with the same input, model and cache state, numerical kernels/execution path, and random draws. Assume the memory wrapper has no side effect other than adding its bias. If that bias is the zero vector at every routed invocation along the trajectory, then every subsequent mathematical state and the output distribution equal those of the base network. Under deterministic numerical execution, the two runs may consequently be bitwise identical.*

Proof. Proceed by induction over the ordered operations in the forward computation. Before the first routed invocation the states agree by assumption. At a routed invocation, the same incoming h and zero bias give the same logits Wh, so the same native router returns the same support and weights and the same frozen experts return the same MoE contribution. The following frozen operation therefore receives the same state. Repeat through the network. Identical output logits imply the same output distribution; with matched random draws, sampling also has the same law and realization. Bitwise equality additionally depends on deterministic kernels and an identical numerical path, which is an implementation condition rather than an algebraic theorem. □

Zero is sufficient rather than necessary: for softmax-based routing, a common per-layer constant is router-equivalent, and other architectures may have further invariances. Conversely, semantic irrelevance of a query does not imply zero bias. The seed constructor must actually return $s = 0$. A full-support softmax slot readout generally does not do so without an explicit gate. An injection method with its own exact OFF branch can satisfy the same conditional equivalence; the property is not unique to routing.

The always-on static module has no unconditional equality guarantee. If a declared continuous output representation, with a specified norm, is locally Lipschitz in the stacked bias with constant $L_{\max}$ on a declared region, then the conditional bound $\|F(x;r) - F(x;0)\| \le L_{\max}\|r\|$ holds there. Freezing the backbone does not prove the premise or convert this norm bound into low semantic interference.

C.6 Conditional architectural compatibility (Proposition C.3)

Proposition C.3 (Conditional architectural compatibility, restated)**.** *Let F_m be a content-augmented model with native route class $\mathcal{F}_{\mathrm{route}}(F_m)$. Adding a routing bias to F_m (i) equals F_m on a zero-bias trajectory under the matched conditions of Theorem 3, and (ii) elsewhere keeps each routed contribution in F_m's native route set at the actual incoming state.*

Proof. (i) is the induction in Theorem 3 applied to F_m. For (ii), replace f_e in Eq. 6 by the expert computations of F_m and apply Proposition 1 recursively. No convex-hull equality is needed for a native unnormalized or sigmoid router. □

The proposition proves only that the mechanisms can be wired together while retaining the zero branch and native-route structure. It does not show that routing retains the content system's benefit, reduces its interference, or beats turning that system down. Those are the three empirical conditions of a useful composition result.

C.7 Algebraic null seed in the graph readout (Proposition C.4)

The deployed graph builder reads at most 120 history texts and 256 tokens per text, retains the 300 most frequent token IDs, adds token–token co-occurrence edges within a five-position forward window, token–expert edges to the eight largest experts in the layer-pooled base router distribution, and expert–expert co-activation edges. Rows are held in a dense array and normalized with a numerical denominator floor; unused rows remain zero. Query token matches place seed mass on the corresponding retained-token rows. These are fixed implementation choices, not learned semantic relevance.

Proposition C.4 (Graph-readout null seed, restated)**.** *Let the dynamic bias be Personalized PageRank on a token–expert graph with transition A, teleport α, and column seed $s(h)$ built from prompt tokens. The implementation initializes $\pi_0 = s$ and runs $T = 30$ iterations:*

$$\pi_{t+1} = \alpha s + (1-\alpha)A^{\top}\pi_t, \qquad q_T = [\pi_T]_{\exp}, \qquad r_u(h) = \beta\nu(q_T).$$

Here A is row-substochastic (zero rows remain zero), $\alpha = .15$, and $\nu(q) = q/\max_e q_e$ for nonnegative q with maximum above 10^{-9}, with $\nu(q) = 0$ otherwise. Then an empty seed yields $r_u(h) = 0$ exactly. Thus any prompt for which the seed construction returns no mass enters the conditional-equivalence branch of Theorem 3. The proposition makes no claim about how often natural unrelated prompts satisfy this condition.

Proof. Let $B = (1-\alpha)A^\top$. Unrolling the implemented recurrence gives

$$\pi_{30} = \left(B^{30} + \alpha \sum_{j=0}^{29} B^j \right) s.$$

Thus the expert-block readout $q_{30} = M_{30}s$ is linear before the implemented max-normalization. Because A is row-substochastic and $\alpha > 0$, the infinite-step limit exists and is $[\alpha(I - (1-\alpha)A^\top)^{-1}s]_{\exp}$; the experiment nevertheless uses the finite $T = 30$ recurrence above. If $s = 0$, every iterate and q_{30} are zero, so $\nu(q_{30}) = 0$ and $r_u = 0$ exactly. A prompt whose tokens place no mass on the history-induced subgraph gives $s = 0$ on that support, so $r_u(h) = 0$ and the base forward computation meets the zero-bias premise. Thus the null seed is an algebraic zero branch. Whether the seed constructor maps a semantically unrelated paraphrase or a keyword-sharing distractor to that branch is a separate detection problem. □

C.8 Final-output margin and interpretation

Final-output margin. Let $m(x) = \mathrm{logit}_{y^\star}(x;0) - \max_{y \neq y^\star} \mathrm{logit}_y(x;0)$ be the base final-logit margin for greedy token $y^\star$. If the *complete network* under routing memory satisfies $\|\Delta\mathrm{logit}(x)\|_\infty \leq \eta$ and $2\eta < m(x)$, then the greedy token is preserved. Indeed, the target can fall by at most η and a competitor can rise by at most η, leaving a positive margin.

This condition must not be conflated with Theorem 1. A router-logit perturbation bound ε does not supply a final-logit bound η without additional layerwise regularity constants, and an unchanged expert set does not imply unchanged mixture weights. In the paper, high-margin controls are therefore empirical measurements rather than consequences of the router gap alone.

Static and prompt-conditioned effects. A static logit shift is the same parameter vector for every prompt, but its network effect can vary across prompts due to routing boundaries, nonlinear experts, and depth. A prompt-conditioned shift additionally varies at the parameter input through $s(h)$. This distinction motivates the co-loading experiment; it does not prove that one parameterization stores a particular number of tasks.

Local existence is not efficacy. For a differentiable scalar objective $J(r)$ with $\nabla J(0) \neq 0$, the directional derivative gives a sufficiently small step that improves J. This elementary local fact neither guarantees that the history estimator finds that direction nor supplies a nonzero effect on any benchmark. Positive memory action and low off-target interference must both be measured, preferably on the same backbone and along a matched-benefit strength curve.