Skill2Mem: A Diagnostic Study of Negative Transfer from Public Skills to LLM Agent Memory

Anonymous ACL submission

Abstract

Public agent ecosystems now distribute reusable `SKILL.md` packages, yet long-term memory systems for LLM agents still rely mostly on fixed, hand-written procedures. We study whether public execution skills can transfer into memory as test-time priors without additional training. We present **Skill2Mem**, a train-free setup that adapts each skill into a compact memory prior and injects selected priors into a retrieval planner under a fixed token budget. Across controlled LongMemEval and LoCoMo evaluations, public skills do not reliably improve the baseline and can introduce structured negative transfer in retrieval planning. Diagnostic analyses show that selected priors concentrate on a small set of execution-oriented skills and that correct-to-incorrect flips outweigh gains under the retrieve-only intervention studied here. The main takeaway is a failure boundary: downstream execution utility does not by itself imply memory alignment.

1 Introduction

Long-term memory has become a central design axis for LLM agents. Benchmarks such as LoCoMo and LongMemEval show that sustained, multi-session interactions require more than larger context windows; systems must decide *what* to store, *how* to consolidate it, and *when* to retrieve or abstain (Maharana et al., 2024; Wu et al., 2024). Recent systems improve these steps with structured storage, hierarchical memory, reflection, or operating-system metaphors (Chhikara et al., 2025; Kang et al., 2025; Tan et al., 2025; Rasmussen et al., 2025; Liu et al., 2026). Yet a common bottleneck remains: most memory pipelines still hard-code a small set of human-written procedures for memory extraction, update, and retrieval.

At the same time, a separate line of practice has emerged outside the memory literature. In public agent ecosystems, *skills* are distributed as reusable procedural packages, typically organized around a `SKILL.md` file with optional scripts and references. SkillsBench shows that curated skills can substantially improve agent pass rates, while self-generated skills often fail to help; notably, focused skills with 2–3 modules outperform comprehensive documentation (Li et al., 2026). These findings suggest that public skill collections encode actionable procedural priors rather than merely verbose documentation.

This paper studies whether such priors transfer to memory. A reasonable but untested hypothesis is that skills useful for downstream execution may also inform upstream memory procedures. We formalize this question as *skill-to-memory transfer*: adapt a public agent skill into a compact prior over memory operations, retrieve a small set of relevant priors at test time, and use them to condition a memory pipeline. This question follows recent work that treats procedural memory as a distinct design target for agents (Fang et al., 2025; Zhang et al., 2026). However, recent procedural-memory systems typically either learn skills through reinforcement learning and self-evolution (Zhang et al., 2026; Yang et al., 2026, 2025) or target execution generalization rather than memory operations (PolySkill, 2025). We instead isolate a train-free regime of *test-time transfer* without learning a controller or updating model parameters.

We instantiate this question through **Skill2Mem**, a deliberately minimal study design in which each public skill is adapted into a

compact memory prior with four fields—trigger, write focus, update rule, and retrieval cues—and a selector injects a small top-k set into a memory system's *retrieval-planning prompt*. Our implementation base is SimpleMem (Liu et al., 2026), a recent train-free system that combines semantic structured compression, online semantic synthesis, and intent-aware retrieval planning. We use MemSkill (Zhang et al., 2026) as a conceptual anchor because it explicitly frames memory operations as skills and provides interpretable skill files, but we do not inherit its training-heavy controller-designer pipeline.

The result is negative, but it points to a specific failure mode. Across controlled LongMemEval and LoCoMo evaluations, no condition reliably exceeds the no-skill baseline; on the shared 300-question LongMemEval subset, curated public priors reduce judge accuracy by 2.7 points. The strongest evidence is diagnostic rather than purely aggregate: selection traces show concentrated retrieval of execution-oriented skills, correctness flips are asymmetric, and confidence gating only recovers baseline accuracy when injection coverage is reduced to a small minority of queries. The issue is *memory alignment*, not just execution utility.

Contributions.

- We formulate *public skills as memory priors* in a train-free, test-time transfer setting built on a strong memory baseline (SimpleMem).
- We show that execution utility does not guarantee memory alignment: in the retrieve-only setting, curated public skills can underperform no-skill, generic, and random controls.
- We contribute an interpretable diagnostic protocol—selection concentration, asymmetric flips, confidence gating, and trace-level case analysis—for studying procedural mismatch in LLM memory systems.

2 Background and Related Work

2.1 Memory systems for LLM agents

Long-term memory benchmarks have highlighted persistent weaknesses of both long-context models and naive retrieval pipelines. LoCoMo introduced very long multi-session dialogues with long-range temporal and causal dependencies (Maharana et al., 2024). LongMemEval decomposes long-term interactive memory into information extraction, multi-session reasoning, temporal reasoning, knowledge updates, and abstention (Wu et al., 2024). MemoryAgentBench broadens the evaluation space toward accurate retrieval, test-time learning, long-range understanding, and selective forgetting in incremental multi-turn settings (Hu et al., 2025).

System designs vary widely. Mem0 dynamically extracts, consolidates, and retrieves salient information, including a graph variant for relational structure (Chhikara et al., 2025). Zep/Graphiti emphasizes a temporal knowledge-graph substrate for dynamic enterprise memory (Rasmussen et al., 2025). Reflective Memory Management combines prospective multi-granularity summarization with retrospective retrieval refinement (Tan et al., 2025). Memory OS of AI Agent introduces a hierarchical storage and update architecture inspired by operating systems (Kang et al., 2025), while MemOS extends the OS metaphor into a broader memory-centric system framework spanning plaintext, activation, and parameter memories (Li et al., 2025). Among train-free systems, SimpleMem is especially relevant because it achieves a strong performance-efficiency frontier via semantic compression, online synthesis, and intent-aware retrieval planning (Liu et al., 2026).

2.2 Skills, procedural memory, and self-evolving agents

The procedural side of agent memory is receiving increasing attention. Memp studies lifelong procedural memory built from trajectories and distinguishes fine-grained instructions from higher-level script-like abstractions (Fang et al., 2025). MemSkill goes one step further by treating extraction, consolidation, and pruning routines as learnable and evolvable memory skills, selected by a controller and refined by a designer (Zhang et al., 2026). AutoSkill similarly proposes experience-driven lifelong learning through self-evolving skill repositories and standardized skill representations (Yang et al., 2026). H2R uses hierarchical hindsight reflection to separate planning-level and execution-level memory in multi-task agents (Ye et al., 2025), and MUSE organizes experience in a hierarchical memory module for self-evolving long-horizon productivity agents (Yang et al., 2025).

These works are conceptually close to ours

but methodologically different. Most optimize or evolve skills through training, reflection loops, or reinforcement learning. Our goal is narrower in scope and remains train-free: to reuse existing external skill resources as *test-time priors* for memory operations without updating the model or learning a controller.

We also note that reinforcement-learning-centric agent frameworks (e.g., OpenClaw-RL) represent a complementary, training-heavy regime for accumulating procedural competence (Wang et al., 2026); Skill2Mem instead focuses on test-time transfer of already-public skills.

2.3 Agentic skills as reusable external priors

In the public Agent Skills ecosystem, skills are portable file-based packages centered on `SKILL.md`, optionally accompanied by scripts, references, and assets. SkillsBench provides the most direct empirical motivation for our work: across 84 evaluated tasks in 11 domains, curated skills improve average pass rate by 16.2 percentage points, self-generated skills do not help on average, and focused 2–3-skill configurations outperform larger collections and comprehensive documentation (Li et al., 2026). PolySkill adds another key insight: separating a skill's abstract goal from its concrete implementation can improve generalization and reuse across websites and tasks (PolySkill, 2025).

These findings suggest that skills may be valuable not only for execution-time guidance, but also as structured external priors for upstream memory operations. Our work evaluates that possibility directly.

2.4 Survey perspectives

Recent surveys reinforce the timeliness of this question. "Rethinking Memory in AI" argues for understanding memory through representation types and atomic operations such as consolidation, updating, indexing, forgetting, retrieval, and compression (Du et al., 2025). "AI Meets Brain" connects cognitive-neuroscience perspectives to LLM agents and explicitly points to skill acquisition as a future direction for memory systems (Liang et al., 2025). Our paper operationalizes this survey-level intuition into a concrete empirical study. Appendix J provides a more detailed positioning table that contrasts train-free memory systems, procedural-memory skill learning, and public skill ecosystems.

3 Problem Setting and Research Questions

We study a memory agent that receives a stream of interactions $H = (u_1, \ldots, u_T)$ and later answers a query q using an external memory store. Existing systems typically rely on a fixed prompt or a small hand-designed set of write/update/retrieve heuristics. We ask whether these fixed procedures can be improved by retrieving a small set of external *skills* $S = \{s_1, \ldots, s_k\}$ and converting them into memory-oriented guidance at test time.

Let $\mathcal{B}$ denote a skill bank (public agent skills or memory-skill markdown files). Each skill is transformed into a compact memory prior $p = A(s)$ containing four operational fields: **trigger** (when it is relevant), **write focus** (what to preserve), **update rule** (how to revise under conflict/updates), and **retrieval cues** (how to search or filter memory). Given a query context q, Skill2Mem retrieves top-k priors under a fixed token budget and injects them into the memory pipeline. In v1, we restrict injection to retrieval planning (retrieve-only), leaving write-side memory construction unchanged.

This scope matters. The present paper studies *retrieval-time priors*, not full write/update memory-skill transfer. We therefore cast the paper around three research questions rather than positive-performance hypotheses:

RQ1: Under a fixed token budget, can curated public skills improve over the `base`, `generic`, and `random` controls?

RQ2: When transfer fails, does the evidence point more strongly to a *memory-alignment gap* than to the mere addition of extra text?

RQ3: In the retrieve-only setting, which memory abilities are most vulnerable to negative transfer from external priors?

4 Method

Skill2Mem is deliberately minimal. It consists of an offline adapter, a selector, and a retrieval-time prompt intervention. This design is sufficient to test whether public skills transfer into memory as *test-time priors* while keeping the underlying memory store and write-side processing fixed. Figure 1 summarizes the study design.

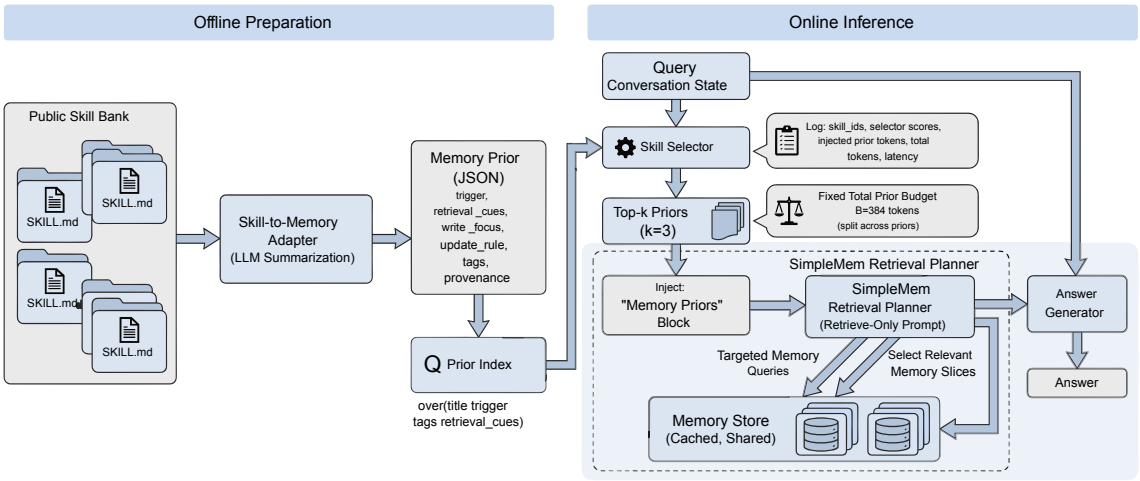

Figure 1: Skill2Mem study design. Public `SKILL.md` skill packages are adapted offline into compact memory priors (trigger, write focus, update rule, retrieval cues) with provenance. At test time, a selector retrieves top-k priors under a fixed token budget and injects them into SimpleMem's retrieval-planning prompt (retrieve-only), while the underlying memory construction remains unchanged.

4.1 Adapting skills into memory priors

Let $\mathcal{B}$ be a skill bank and $s \in \mathcal{B}$ a skill package (a folder centered on `SKILL.md`). Skill2Mem maps each s to a *memory prior* $p = A(s)$ with four operational fields: **trigger** (when relevant), **retrieval cues** (what to query/filter), **update rule** (how to revise under conflict or updates), and **write focus** (what should have been preserved at write time). Each prior also stores provenance (source, commit, license) and lightweight tags. The adapter is an offline normalization step rather than a learned component. Its role is to convert heterogeneous skills into a comparable schema so that the study tests *transfer* rather than raw document length or formatting. Exact prompt wording, schema validation, and provenance handling are deferred to Appendix C.

Budget control. To separate "procedural alignment" from "adding more text", we enforce a fixed total prior-token budget B (384 tokens). For a top-k set of priors, we allocate at most $\lfloor B/k \rfloor$ tokens per prior (after formatting) and truncate the final concatenated block to B tokens. This removes length advantages from longer `SKILL.md` documents (details in Appendix C).

4.2 Selecting priors under a fixed budget

Given a query q, we retrieve top-k adapted priors by matching over the adapted prior text. We intentionally keep k small to limit prompt overhead and to mirror SkillsBench's finding that a few focused skills can outperform comprehensive documentation (Li et al., 2026). Appendix A gives the exact selector instantiation used in our evaluations.

Formal view. Let $\mathcal{P}_{\mathcal{B}}$ denote the adapted prior set from bank $\mathcal{B}$, and let $\mathrm{sim}(q, p)$ be the selector similarity. Skill2Mem selects $P_k(q) = \mathrm{TopK}_{p \in \mathcal{P}_{\mathcal{B}}} \mathrm{sim}(q, p)$ and injects a formatted block $I(P_k(q), B)$ into the retrieval planner, where $I(\cdot, B)$ enforces the fixed token budget B. We view this as supplying a test-time procedural prior to the planner without changing the underlying model or write-side memory.

4.3 Intervention scope: retrieval planning only

We inject priors into retrieval planning because this stage determines *which* memory slices are searched and *how* evidence is aggregated. This is a natural point of intervention for SimpleMem, which explicitly includes intent-aware retrieval planning (Liu et al., 2026). We insert

a compact `Memory Priors:` block into that planner prompt and leave write-side processing unchanged. This restricted intervention isolates retrieval-time priors from changes to memory construction.

5 Experimental Setup

The main paper reports only the setup details needed to interpret the scientific claim; model IDs, prompt text, subset construction, and reproducibility details are deferred to Appendix A and Appendix I.

Benchmarks. We use LongMemEval as the primary benchmark because it targets the memory abilities most exposed to retrieval-time priors: information extraction, multi-session reasoning, temporal reasoning, knowledge updates, and abstention (Wu et al., 2024; Feng et al., 2024). We use LoCoMo as a secondary consistency check because it stresses very long conversational memory with event and temporal structure in a different interaction format (Maharana et al., 2024).

Baseline and controls. The literature baseline is SimpleMem (Liu et al., 2026), used without external priors. Controlled variants keep the same retrieve-only intervention and change only the injected prior block: a budget-matched generic guideline, random priors from the same bank, curated public priors adapted from a curated public skill collection, noisy public priors adapted from a broader community collection, and memory-native priors adapted from MemSkill (Zhang et al., 2026). Bank composition and licensing are summarized in Appendix B.

Evaluation. All conditions keep write-side memory construction fixed and vary only the retrieval-time prior block. For LongMemEval, we report judge accuracy as the primary metric and exact match (EM) as a stricter secondary metric. For LoCoMo, we report the official category-aware QA score. Uncertainty estimates, flip analyses, and efficiency statistics are reported only in the appendix.

6 Results

6.1 Main results on LongMemEval

Table 1 reports the main LongMemEval result on the shared 300-question evaluation subset. No condition reliably exceeds the no-skill baseline. The largest directional change is negative: curated public priors reduce judge accuracy from 0.760 to 0.733, a 2.67-point drop relative to `base`.

RQ1 is therefore not supported in this setting: curated public priors do not outperform `base`, `generic`, or `random`. Token budget alone also does not explain the result: `generic` trails `base`, whereas `random` and `noisy` remain effectively tied with it. EM is flatter than judge accuracy, so the core finding is not uniform metric collapse but a more specific pattern of judge-side degradation, asymmetric flips, and unstable retrieval traces under curated priors.

We therefore read these results as a diagnosis rather than as a standalone performance comparison. The interval estimates remain wide at this scale, but the qualitative pattern is consistent: curated shows the largest average drop, the most negative flip balance, and the strongest concentration in a few execution-oriented priors. Section 7 develops this pattern into a mechanism-level diagnosis.

6.2 LoCoMo as a consistency check

LoCoMo shows the same pattern with weaker separation. On the 200-question evaluation subset, all retrieve-only variants remain within 0.005 of the SimpleMem baseline (0.338), including curated public priors at 0.338. We therefore use LoCoMo as an external consistency check: it offers no evidence of stable positive transfer, but it also does not show the same magnitude of degradation observed on LongMemEval. The full table is reported in Appendix H.

7 Analysis

Negative transfer is concentrated, not diffuse. Figure 2 provides the main quantitative diagnostic evidence. The left panel shows a coverage–accuracy tradeoff under confidence gating: curated public priors recover `base` accuracy only at τ=0.125, which leaves just 4.3% of queries injected. MemSkill shows a similar pattern and reaches `base` only at 1.0% coverage. The right panel shows the complementary failure mode: top-1 selection is highly concentrated in a few skills. In the curated bank, `algorithmic-art` alone accounts for 39.7% of top-1 queries and `doc-coauthoring` for 23.0%, whereas MemSkill spreads mass over memory-native operators such as `delete`, `insert`, and `capture-temporal-context`. This concen-

Condition	Judge	Δ (pp)	95% CI	EM	Pos/Neg flips
SimpleMem (Liu et al., 2026)	0.760	+0.00	—	0.247	—
Generic guideline	0.747	−1.33	[−4.00, +1.33]	0.233	6/10
Random priors	0.763	+0.33	[−2.33, +3.00]	0.240	8/7
Curated public priors	0.733	−2.67	[−5.33, +0.00]	0.243	5/13
Noisy public priors	0.763	+0.33	[−1.67, +2.33]	0.260	5/4
MemSkill priors (Zhang et al., 2026)	0.740	−2.00	[−4.33, +0.33]	0.243	4/10

Table 1: LongMemEval results on the shared 300-question evaluation subset. Judge deltas and 95% paired-bootstrap intervals are computed relative to `base`; flips count incorrect→correct / correct→incorrect transitions. Detailed setup and efficiency statistics are reported in the appendix.

tration is consequential: `algorithmic-art` appears in 12 of the 13 curated correct→incorrect flips (Appendix Table 7).

A representative failure trace. Figure 3 connects the corpus-level pattern to a concrete mechanism. The left panel summarizes the most frequently retrieved skills under curated and memory-native banks, making the bank mismatch visually explicit. The right panel then shows how that mismatch manifests in a single question: the query requires temporal comparison across memory evidence; without priors, the planner retrieves enough support to answer correctly; with curated public priors, it is steered toward broad execution-style cues and ends in abstention instead. The figure shows that the observed degradation can be traced to a specific mechanism rather than treated as opaque noise.

The drop concentrates in memory-heavy question types. RQ3 also receives a specific answer. The curated drop is concentrated in multi-session and temporal reasoning rather than spread uniformly across the benchmark (Appendix E). On the shared 300-question subset, curated falls from 0.671 to 0.605 on multi-session questions (76 items) and from 0.630 to 0.593 on temporal reasoning (81 items), while knowledge-update questions remain stable or improve (0.848 to 0.870). This pattern is consistent with a retrieval-planning mismatch: cross-session aggregation and time normalization are precisely the abilities most exposed to bad procedural priors at retrieval time.

Evidence for an alignment gap. The evidence favors an *alignment-gap* explanation over a simple "more text hurts" explanation. Same-budget generic text is mildly harmful, but curated priors are worse and fail in a more structured way: they are selected with low confidence, concentrate in a small number of execution-oriented skills, and produce a more negative flip balance than `generic`. Together, these observations suggest that execution-oriented public skills cannot be treated as directly transferable memory procedures.

8 Discussion

Scope of the contribution. The main contribution of Skill2Mem is not a stronger memory system, but a controlled test of a plausible transfer hypothesis. In the studied setting—SimpleMem, retrieve-only injection, TF–IDF selection, and controlled evaluation subsets—public skills do *not* behave like reliable memory priors. The negative result is useful because the failures are structured: they concentrate in a few selected skills, a few question types, and a narrow portion of planner behavior rather than appearing as diffuse noise.

Claims outside scope. We do not claim that public skills are broadly harmful to memory. We also do not claim that curated public banks are inherently worse than noisy ones, because the current curated/noisy comparison is confounded by bank size (18 vs. 100 skills) and domain diversity. Finally, we do not claim that memory-native priors are ineffective in general: the present study only intervenes at retrieval planning time, so write-time normalization and update policies remain untested.

Implications for future systems. External priors remain attractive because they are controllable, inspectable, and train-free. In our setting, though, they need *memory-aware alignment* before they help. The next steps are mostly methodological: test size-matched banks, strengthen selectors, and move beyond retrieve-only prompting to interventions that can change the memory representation itself. In that sense, Skill2Mem uses public skills as a controlled setting for studying when external

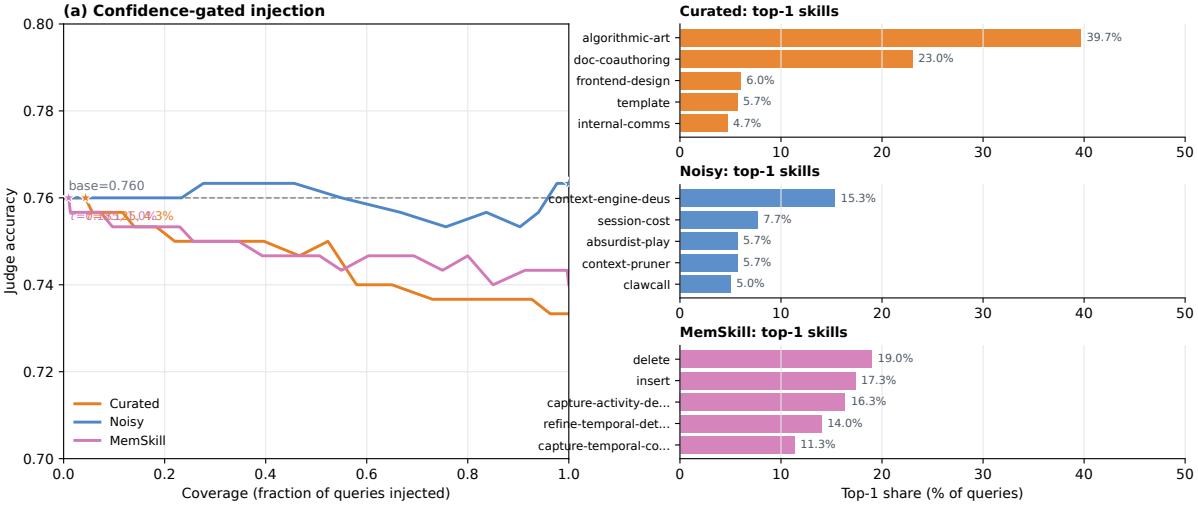

Figure 2: Quantitative diagnosis of negative transfer on the shared 300-question LongMemEval evaluation subset. Left: confidence-gated injection varies a threshold on top-1 TF–IDF similarity and reveals a coverage–accuracy tradeoff relative to `base`. Right: most frequent top-1 retrieved skills, reported as % of queries for each bank. Curated public skills show both stronger concentration and sharper gating requirements than the budget-matched controls.

procedures transfer to memory and when they do not.

9 Limitations

Our evidence comes from controlled evaluations on fixed subsets, so the paper should be read as a diagnostic study rather than a final benchmark verdict. Full-scale runs, multi-seed estimates, and open-model replications remain future work. We also focus on a single intervention point (retrieve-only injection into retrieval planning) and a single primary base system (SimpleMem); other memory pipelines and write-time interventions may exhibit different transfer dynamics.

Risks. Our skill banks are also limited. The curated public bank is small (18 skills) and reflects a productivity-oriented ecosystem rather than memory alignment; the noisy bank is larger (100 skills) and not size-matched to the curated bank. Although we control the injected prior budget, bank size and domain diversity may still affect selector behavior. Finally, the selector is deliberately simple (TF–IDF similarity) and the adapter compresses heterogeneous skill styles into a uniform schema; both choices may contribute to the observed alignment gap. Public skill sources are heterogeneous in quality, safety, and licensing, so provenance tracking and filtering are necessary before releasing or deploying derived priors.

10 Ethical Considerations

External skills can encode biases, unsafe instructions, or domain-specific assumptions. Reusing them as memory priors therefore requires at least three safeguards: provenance and license logging, filtering of clearly unsafe skills and high-risk domains, and auditing of failure modes where a prior induces incorrect overwrites, unsafe retrieval suggestions, or privacy-sensitive storage patterns. More broadly, importing industrial agent resources into academic memory benchmarks raises questions about transparency and redistributability. We therefore support releasing adapted priors, selection traces, and prompts only where licensing permits, and otherwise releasing aggregate statistics and reconstruction scripts for rebuilding banks from upstream sources.

11 Conclusion

We studied whether public agent skills transfer into long-term memory systems as *test-time* priors without any training. In the studied retrieve-only setting, no condition reliably improves over the SimpleMem baseline, and curated public priors produce the clearest LongMemEval degradation. The contribution is diagnostic rather than performance-driven. Execution-oriented public skills can be retrieved often enough to alter retrieval plans, yet still be misaligned with the memory operations that the query actually needs.

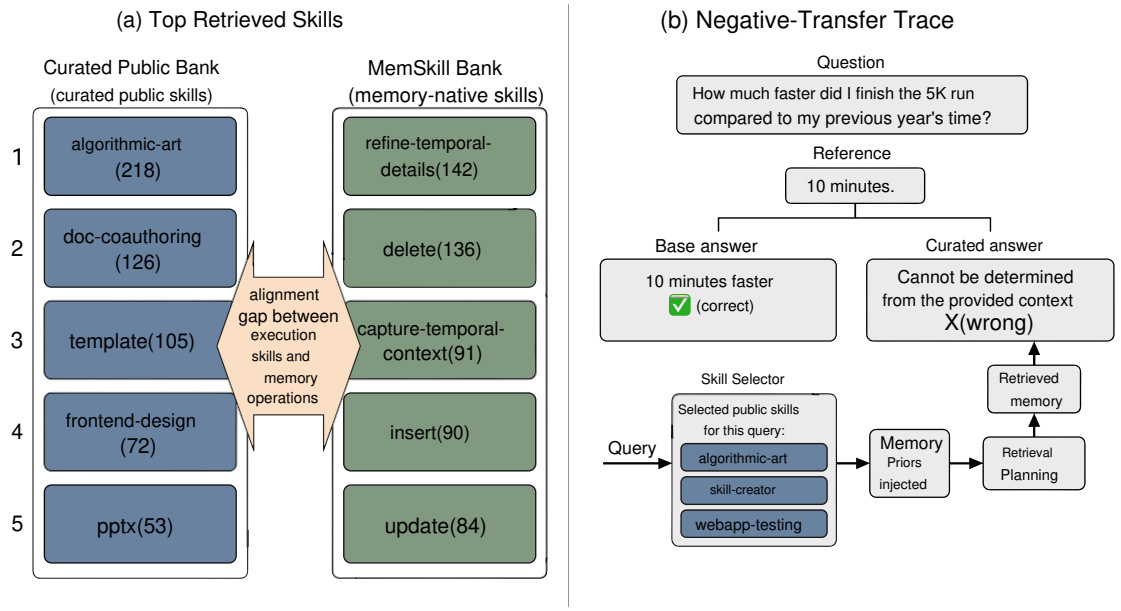

Figure 3: Representative mechanism of negative transfer. Left: the curated public bank is dominated by a few broad execution-oriented skills, whereas the MemSkill bank emphasizes memory-native operations. Right: on a LongMemEval example requiring temporal comparison, curated public priors redirect retrieval planning away from the needed memory operation and turn an answerable case into an abstention.

More broadly, the study suggests that downstream execution utility is not the same as *memory alignment* for upstream memory procedures. Because priors remain explicit and inspectable, the failure mode can be analyzed directly by tracing which skill was selected, how concentrated the selector becomes, where correctness flips occur, and how gating changes the coverage–accuracy tradeoff. Public skills therefore remain a promising but unrefined external resource. Progress is more likely to come from memory-specific alignment and stronger intervention points than from injecting additional skill text into an already strong planner prompt.

References

Yutao Yang, Junsong Li, Qianjun Pan, Bihao Zhan, Yuxuan Cai, Lin Du, Jie Zhou, Kai Chen, Qin Chen, Xin Li, Bo Zhang, and Liang He. 2026. AutoSkill: Experience-Driven Lifelong Learning via Skill Self-Evolution. arXiv preprint arXiv:2603.01145.

Yiming Du, Wenyu Huang, Danna Zheng, Zhaowei Wang, Sebastien Montella, Mirella Lapata, Kam-Fai Wong, and Jeff Z. Pan. 2025. Rethinking Memory in LLM based Agents: Representations, Operations, and Emerging Topics. arXiv preprint arXiv:2505.00675.

Runnan Fang, Yuan Liang, Xiaobin Wang, Jialong Wu, Shuofei Qiao, Pengjun Xie, Fei Huang, Huajun Chen, and Ningyu Zhang. 2025. Memp: Exploring Agent Procedural Memory. arXiv preprint arXiv:2508.06433.

Shicheng Ye, Chao Yu, Kaiqiang Ke, Chengdong Xu, and Yinqi Wei. 2025. H2R: Hierarchical Hindsight Reflection for Multi-Task LLM Agents. arXiv preprint arXiv:2509.12810.

Yuanzhe Hu, Yu Wang, and Julian McAuley. 2025. Evaluating Memory in LLM Agents via Incremental Multi-Turn Interactions. arXiv preprint arXiv:2507.05257.

Jiazheng Kang, Mingming Ji, Zhe Zhao, and Ting Bai. 2025. Memory OS of AI Agent. arXiv preprint arXiv:2506.06326.

Zhiyu Li, Chenyang Xi, Chunyu Li, et al. 2025. MemOS: A Memory OS for AI System. arXiv preprint arXiv:2507.03724.

Jiafeng Liang, Hao Li, Chang Li, et al. 2025. AI Meets Brain: Memory Systems from Cognitive Neuroscience to Autonomous Agents. arXiv preprint arXiv:2512.23343.

Haozhen Zhang, Quanyu Long, Jianzhu Bao, Tao Feng, Weizhi Zhang, Haodong Yue, and Wenya Wang. 2026. MemSkill: Learning and Evolving Memory Skills for Self-Evolving Agents. arXiv preprint arXiv:2602.02474.

Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. 2024. Evaluating Very Long-Term Conversational Memory of LLM Agents. arXiv preprint arXiv:2402.17753.

Yinjie Wang, Xuyang Chen, Xiaolong Jin, Mengdi Wang, and Ling Yang. 2026. OpenClaw-RL: Train Any Agent Simply by Talking. arXiv preprint arXiv:2603.10165.

Simon Yu, Gang Li, Weiyan Shi, and Peng Qi. 2025. PolySkill: Learning Generalizable Skills Through Polymorphic Abstraction. arXiv preprint arXiv:2510.15863.

Preston Rasmussen, Pavlo Paliychuk, Travis Beauvais, Jack Ryan, and Daniel Chalef. 2025. Zep: A Temporal Knowledge Graph Architecture for Agent Memory. arXiv preprint arXiv:2501.13956.

Jiaqi Liu, Yaofeng Su, Peng Xia, Siwei Han, Zeyu Zheng, Cihang Xie, Mingyu Ding, and Huaxiu Yao. 2026. SimpleMem: Efficient Lifelong Memory for LLM Agents. arXiv preprint arXiv:2601.02553.

Xiangyi Li, Wenbo Chen, Yimin Liu, et al. 2026. SkillsBench: Benchmarking How Well Agent Skills Work Across Diverse Tasks. arXiv preprint arXiv:2602.12670.

Zhen Tan et al. 2025. In Prospect and Retrospect: Reflective Memory Management for Long-term Personalized Dialogue Agents. arXiv preprint arXiv:2503.08026.

Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. 2024. LongMemEval: Benchmarking Chat Assistants on Long-Term Interactive Memory. arXiv preprint arXiv:2410.10813.

Shangbin Feng, Weijia Shi, Yike Wang, Wenxuan Ding, Vidhisha Balachandran, and Yulia Tsvetkov. 2024. Don't Hallucinate, Abstain: Identifying LLM Knowledge Gaps via Multi-LLM Collaboration. arXiv preprint arXiv:2402.00367.

Cheng Yang, Xuemeng Yang, Licheng Wen, et al. 2025. Learning on the Job: An Experience-Driven Self-Evolving Agent for Long-Horizon Tasks. arXiv preprint arXiv:2510.08002.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. 2025. Mem0: Building Production-Ready AI Agents with Scalable Long-Term Memory. arXiv preprint arXiv:2504.19413.

A Experimental protocol and baselines

We report controlled subset evaluations rather than full-benchmark sweeps. All LongMemEval conditions are evaluated on the same 300-question subset, and all LoCoMo conditions are evaluated on the same 200-example subset. Sampling, random controls, and paired comparisons use fixed seeds so that every condition is scored on identical items.

System settings. Unless otherwise stated, all reported runs use retrieval-only intervention, a fixed prior budget $B = 384$, $k = 3$ injected priors, and deterministic decoding. The underlying memory system is SimpleMem (Liu et al., 2026). Table 2 consolidates the exact models, banks, and recorded outputs used throughout the study.

Evaluation outputs. For LongMemEval, we record exact match and judge accuracy under the yes/no protocol of Wu et al. (2024). For LoCoMo, we record the official category-aware QA score. Efficiency measures (tokens and latency) are kept out of the main paper and reported only in the appendix.

B Skill bank statistics and licensing

Table 3 summarizes the three skill banks used in our LongMemEval evaluation subset study. Token counts are computed over each skill's markdown text using a cl100k-style tokenizer. Licenses are best-effort detections from source-level license files.

Skill-document length distribution. Figure 4 visualizes the token-length distributions of raw `SKILL.md` documents in each bank. Public skills are often thousands of tokens long, while our prior injection is capped at a fixed budget B=384 tokens; this gap motivates our budgeted truncation scheme and ensures comparability across banks.

C Adapter prompt and prior schema

Prior schema. Each skill is adapted into a compact structured prior containing trigger, retrieval, write, and update fields plus provenance and license metadata. We validate adapted priors against a strict schema before reuse across runs.

Adapter prompt (excerpt).

```
You are adapting a public agent skill
(SKILL.md) into a compact memory prior
for an LLM memory system at test time.

Return one JSON object with keys:
  trigger
  retrieval_cues
  write_focus
  update_rule
  tags
  source
  license

Constraints:
```

Item	Choice	Notes / reference
Base system	SimpleMem	Train-free memory pipeline with intent-aware retrieval planning (Liu et al., 2026)
Primary benchmark	LongMemEval	Interactive long-term memory benchmark with five ability categories (Wu et al., 2024)
Secondary benchmark	LoCoMo	Very long conversational memory benchmark (Maharana et al., 2024)
Agent model	`gpt-5.4-high`	Used for generation in all pilots; fixed across conditions
Adapter model	`gpt-5.2-low`	Used for one-shot prior adaptation in the reported configuration
Judge model	`gpt-5.2-low`	Used for LongMemEval yes/no judging
Subset recipe	`seed=42`, deterministic shared subsets	Shared 300-question LongMemEval subset; shared 200-example LoCoMo subset
Curated public bank	curated public skill collection	Curated public bank used in the study
Noisy public bank	broader public skill collection	Larger public-skill control bank
Memory-native bank	MemSkill skill files	Research-native memory-operation source (Zhang et al., 2026)
Controls	Generic guideline; random priors	Same-budget non-skill and random-selection controls introduced in this paper
Decoder / selector	temperature 0; TF–IDF top-k matching	Shared across conditions; retrieval-side only
Logged outputs	per-example selections, selector scores, efficiency summaries, case annotations	Used only for diagnosis and appendix analyses

Table 2: Experimental protocol used throughout the reported subset evaluations. The supplementary materials summarized in Section I provide the exact reconstruction recipe for the reported configuration.

Bank	N	Mean tok	Median	Min–Max	License (best-effort)
Curated public	18	2354	2031	28–7322	12 Apache-2.0; 4 per-skill; 2 unknown
Noisy public	100	1516	1019	16–12875	100 MIT
MemSkill (research-native)	16	140	150	81–182	16 Apache-2.0

Table 3: Skill bank statistics and license breakdowns for the LongMemEval evaluation subset.

```
- Be concrete and operational.
- Do not copy long text.
- Prefer memory procedures:
  temporal normalization,
  conflict handling,
  session boundaries,
  provenance.

SKILL.md:
<...skill markdown...>
```

Proposition (budget fairness). For any query, Skill2Mem injects at most B prior tokens, and therefore does not advantage conditions with longer skill documents. *Proof.* Each retrieved prior is formatted and truncated to at most $\lfloor B/k \rfloor$ tokens; the concatenated block is then truncated again to B tokens.

Injection format. At test time, we format each prior into a single compact line and inject a top-k set into the retrieval-planning prompt under a fixed token budget:

```
- <skill_id>: trigger=...
  retrieval=... | update=...
  write=... | tags=...
```

Generic guideline baseline. The `generic` condition injects a budget-matched retrieval guideline instead of skill-specific priors. The injected text is truncated to the same total budget (B=384) to isolate the effect of *procedural content* rather than length:

```
Goal: retrieve minimal relevant memory.

Procedure:
- Parse into atomic needs.
- Generate 1--3 targeted queries.
- Treat time as first-class.
- Resolve conflicts using timestamps.
- Abstain when memory is missing.
```

D Uncertainty, flips, and selector confidence

Paired bootstrap uncertainty (judge accuracy). Table 4A reports paired bootstrap uncertainty for per-condition judge-accuracy deltas relative to `base`, computed over the 300 shared questions (20,000 bootstrap resamples; seed 0). We report percentile 95% intervals.

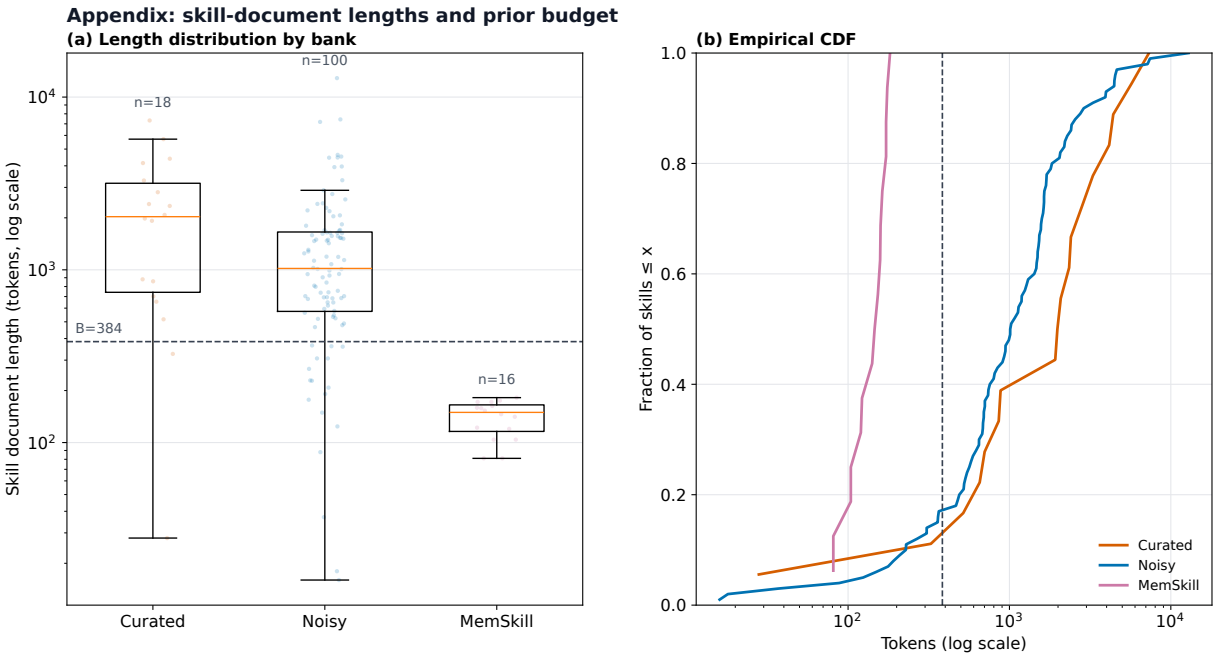

Figure 4: Skill-document lengths by bank (log scale) and empirical CDFs. The dashed line marks the fixed prior-token budget B=384 used in all conditions.

Correctness flips vs `base` (judge labels). Table 4B reports correctness flips relative to `base` on the 300 shared LongMemEval questions. "Positive" flips are incorrect-to-correct; "negative" flips are correct-to-incorrect.

Selector confidence statistics. Table 4C reports selector confidence statistics for conditions that retrieve priors using TF–IDF similarity. We report the top-1 similarity score per query.

E LongMemEval breakdown by question type

Table 5 reports judge-accuracy breakdowns for the shared 300-question LongMemEval evaluation subset.

F Selection traces and case studies

The selector-level quantitative diagnostics are kept in main Figure 2, and the main text also includes one representative trace (Figure 3). The appendix therefore focuses on supporting counts and a small set of additional cases.

Most frequently retrieved skills. Table 6 shows the most frequently selected skills in the shared 300-question LongMemEval evaluation subset. Counts are over all selected slots (300 questions $\times$ k=3).

Skills associated with negative flips. To localize failure modes, we count which retrieved skills appear on *correct→incorrect* flips relative to `base` (judge labels). In the curated public bank, a small number of broad execution skills dominate negative flips, consistent with an alignment gap between public execution skills and memory procedures. In MemSkill, negative flips involve memory-native operators, suggesting a different failure mode (prompt-policy interference).

Representative curated-bank cases. Table 8 summarizes four representative cases drawn from the stored improve/degrade lists. We keep the appendix focused on a few interpretable examples here and defer the full case inventory to the supplementary materials.

G LoCoMo and efficiency

Table 9 reports mean token usage by stage on LongMemEval. The fixed prior-token budget primarily increases retrieval-planning stages, while answer-generation tokens remain comparable.

H LoCoMo results

Table 10 reports the LoCoMo subset evaluation (200 QA; seed 42; retrieval-only; k = 3; total prior budget 384). Unlike LongMemEval, LoCoMo shows essentially no separation among con-

(A) Paired bootstrap Δ vs base

Condition	Δ (pp)	95% CI
Generic	-1.33	$[-4.00, +1.33]$
Random	$+0.33$	$[-2.33, +3.00]$
Curated	-2.67	$[-5.33, +0.00]$
Noisy	$+0.33$	$[-1.67, +2.33]$
MemSkill	-2.00	$[-4.33, +0.33]$

(B) Correctness flips vs base

Cond.	Pos	Neg	Net
Generic	6	10	-4
Random	8	7	$+1$
Curated	5	13	-8
Noisy	5	4	$+1$
MemSkill	4	10	-6

(C) Selector confidence (top-1 sim.)

Bank	Mean	Med	P90	% $<$ 0.05
Curated	0.070	0.067	0.114	27.0
Noisy	0.067	0.063	0.099	24.3
MemSkill	0.095	0.092	0.144	8.7

Table 4: LongMemEval evaluation-subset summary statistics for (A) paired bootstrap uncertainty of judge-accuracy deltas, (B) correctness flips relative to `base`, and (C) selector confidence (top-1 TF–IDF similarity).

Type	N	Base	Gen.	Rand.	Cur.	Noisy	Mem
knowledge-update	46	0.848	0.826	0.891	0.870	0.891	0.870
multi-session	76	0.671	0.671	0.645	0.605	0.645	0.618
single-session-assistant	40	1.000	1.000	1.000	1.000	1.000	1.000
single-session-preference	15	0.467	0.400	0.467	0.400	0.533	0.333
single-session-user	42	0.952	0.952	0.976	0.952	0.952	0.952
temporal-reasoning	81	0.630	0.605	0.630	0.593	0.630	0.617

Table 5: LongMemEval breakdown by question type on the shared 300-question evaluation subset (judge accuracy). The agent and judge models are fixed across conditions.

Curated (public)	
`algorithmic-art`	218
`doc-coauthoring`	126
`template`	105
`frontend-design`	72
`pptx`	53
`canvas-design`	44
`webapp-testing`	34
`claude-api`	33
`web-artifacts-builder`	33
`internal-comms`	31
MemSkill (memory-native)	
`refine-temporal-details`	142
`delete`	136
`capture-temporal-context`	91
`insert`	90
`update`	84
`capture-activity-details`	84
`track-object-movements`	39
`capture-entity-nuances`	37
`track-object-location`	36
`handle-entity-relationships`	29

Table 6: Most frequently selected skills on the LongMemEval evaluation subset (top-10 by count; 900 total selected slots).

Curated negative flips (N=13)	
`algorithmic-art`	12
`template`	6
`doc-coauthoring`	3
`webapp-testing`	3
`docx`	2
`canvas-design`	2
`internal-comms`	2
`pptx`	2
`web-artifacts-builder`	1
`mcp-builder`	1
MemSkill negative flips (N=10)	
`capture-activity-details`	5
`delete`	5
`refine-temporal-details`	5
`insert`	4
`update`	4
`capture-entity-nuances`	1
`noop`	1
`track-object-movements`	1
`handle-entity-relationships`	1
`track-object-location`	1

Table 7: Top retrieved skills on correct→incorrect flips vs `base` (top-10 by count; LongMemEval evaluation subset; judge).

ditions at this scale; efficiency overhead is still visible.

I Reproducibility summary

The supplementary materials accompanying this paper contain four components needed to reconstruct the reported runs and inspect their behavior: (i) configuration snapshots with exact model IDs, seeds, and prior-budget settings; (ii) bank provenance records with revision and license metadata; (iii) per-example selection records, predictions, and judge outcomes; and (iv) aggregated reports including uncertainty estimates, type breakdowns, and improve/degrade case lists.

Exact snapshot. For the main LongMemEval study, the reported configuration fixes the agent model to `gpt-5.4-high`, the adapter and judge model to `gpt-5.2-low`, the shared seed to 42,

Dir.	QID	Type	Query / reference	Base → curated	Selected skills
Degrade	`61f8c8f8`	multi-session	5K improvement vs. previous year; ref: 10 minutes	"10 minutes faster" → "Cannot be determined"	`algorithmic-art`, `skill-creator`, `webapp-testing`
Degrade	`a9f6b44c`	multi-session	bikes serviced/planned in March; ref: 2	"2 bikes" → "1 bike"	`mcp-builder`, `algorithmic-art`, `template`
Improve	`7401057b`	knowledge-update	Hilton free-night stays; ref: two stays	"1 free night" → "Two free-night stays"	`web-artifacts-builder`, `algorithmic-art`, `template`
Improve	`edced276`	multi-session	total travel days in Hawaii + NYC; ref: 15 days	"Cannot determine total" → "15 days total"	`xlsx`, `doc-coauthoring`, `algorithmic-art`

Table 8: Representative curated-bank case snapshots from the LongMemEval evaluation subset. Full improve/degrade case lists are included in the supplementary materials.

Condition	Plan info req.	Plan targeted queries	Analyze query	Answer gen.
Base	714	878	388	3841
Generic	1131	1237	395	3793
Random	1082	1224	391	3733
Curated	1090	1235	396	3771
Noisy	1104	1227	393	3781
MemSkill	1098	1218	390	3737

Table 9: Mean tokens by stage on the LongMemEval evaluation subset. The agent model is fixed across conditions.

Cond.	Score	Tok.	Sec.
Base	0.338	3,343	32.88
Generic	0.333	4,096	41.29
Random	0.337	4,070	54.38
Curated	0.338	4,068	53.92
MemSkill	0.337	4,078	40.30

Table 10: LoCoMo results on the shared 200-question subset (retrieval-only; $k = 3$; B=384).

the injected-prior count to k=3, and the total prior budget to B=384. These records are sufficient to reproduce the condition-level tables and inspect which priors were selected for each query.

Secondary observations. Taken together, the appendix tables reinforce two observations that are useful but not central enough for the main paper. First, the extra compute is concentrated in retrieval-planning stages rather than answer generation, which is consistent with the narrow intervention studied here. Second, LoCoMo remains essentially flat across conditions at this scale, suggesting that the negative-transfer signal is strongest when the benchmark explicitly stresses multi-session reasoning, temporal normalization, and update-sensitive retrieval.

J Positioning overview

Table 11 provides a positioning view that contrasts train-free memory systems, procedural-memory skill learning, and public skill ecosystems. This comparison makes the paper's scope precise. Unlike MemSkill or AutoSkill, we do not learn or evolve memory skills. Unlike SkillsBench, we do not study execution-time skill utility. Unlike SimpleMem, Mem0, or Memory OS, we do not propose a new memory architecture. The contribution instead sits at the intersection of these lines: a train-free memory base, explicit public skill resources, and a diagnostic evaluation of whether procedural transfer succeeds at test time.

Work	Year	Core idea	Train-free?	Main benchmarks	Relevance to this paper
LoCoMo	2024	Very long multi-session conversational benchmark with temporal/event structure	–	LoCoMo	Primary benchmark for long-range memory and event reasoning
LongMemEval	2024/25	Interactive memory benchmark with five abilities and a 3-stage design lens	–	LongMemEval	Primary benchmark for update/abstention and interactive memory
Mem0	2025	Dynamic extraction, consolidation, retrieval; graph variant	Yes	LoCoMo	Strong practical baseline in train-free memory design
RMM	2025	Prospective and retrospective reflection for dialogue memory	No/Hybrid	LongMemEval	Relevant retrieval-refinement comparator; more complex than our target
Memory OS	2025	Hierarchical storage/update architecture inspired by OS	Mostly yes	LoCoMo	Shows value of structured memory operations without skill transfer
Memp	2025/26	Learnable lifelong procedural memory from trajectories	No/Hybrid	ALFWorld, TravelPlanner	Closest procedural-memory framing; useful conceptual neighbor
H2R	2025	Hierarchical hindsight reflection for planning/execution memory	No	ALFWorld, PDDLGame	Supports stage decomposition and hierarchical memory intuition
PolySkill	2025/26	Decouples skill goal from implementation for generalization	No	Mind2Web	Motivates abstraction when adapting public skills
MemSkill	2026	Memory operations as learnable/evolvable skills	No	LoCoMo, LongMemEval	Main conceptual anchor; source of research-native skill files
AutoSkill	2026	Experience-driven lifelong learning via self-evolving skill repository	No/Hybrid	Multi-domain	Shows skill standardization and cross-agent reuse potential
SkillsBench	2026	Evaluates how curated skills help agents across domains	–	84 tasks, 11 domains	Main motivation that skill curation and granularity matter
SimpleMem	2026	Train-free semantic compression, synthesis, retrieval planning	Yes	LoCoMo	Main implementation substrate for Skill2Mem

Table 11: Positioning of the proposed study. We deliberately target the narrow intersection of *train-free memory systems* and *external skill priors*.