EXECUTABLE MEMORY WITH AUTHENTICATED RUNTIME ADMISSION FOR LONG-TERM AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-term agent memory becomes operational when stored instructions govern tool use. Retrieval-based systems ask language models both to select relevant memory and to reconstruct its authorized effect, coupling semantic judgment with security-sensitive execution. We introduce Executable Memory with Authenticated Runtime Admission (EMARA), a systems architecture that represents registered, authority-bearing memory as typed programs and mediates their use at runtime. EMARA separates learned semantic nomination from trusted, state-bound admission. The model nominates public program handles; a reference monitor evaluates current-state predicates and releases only actions bound to authenticated certificates. We evaluate EMARA on MemBind-Bench, a long-context benchmark spanning web policies, workspace boundaries, secret handling, atomic program composition, and time-scoped knowledge use. Comparisons with textual, retrieval, external-memory, and executable baselines show that separating nomination from admission enables reliable, state-consistent execution across diverse memory settings. Controlled studies isolate semantic nomination, stored program content, action realization, certificate verification, request wording, and backbone choice. Together, the results identify runtime admission, rather than retrieval alone, as the mechanism that enables authorized execution from long-term agent memory.

1 INTRODUCTION

Agents preserve preferences, credentials, and operating rules as text. Acting on such memory requires selecting a rule, checking whether it applies now, and constructing an authorized action. Search improves access to stored text. Model generation still entangles these three decisions Vu et al. (2024). Long histories and near-miss distractors amplify the resulting errors.

EMARA separates selection from authority. A deployer registers an authority-bearing rule as a typed program; descriptive memory remains textual. The model nominates a public handle as evidence. A trusted runtime supplies authority by probing current state and permitting only action fields bound to a valid certificate.

We contribute (1) EMARA, an authenticated runtime-admission architecture; (2) MemBind-Bench, a five-family, 5,532-case benchmark; and (3) aligned controls that isolate nomination, program content, execution, verification, opaque authority, error containment, and memory lifetime. EMARA improves task success by 31.45 points over the strongest non-oracle baseline and reduces unsafe actions from 23.08% to 0.00%. EMARA names the architecture, and MemBind-Bench names the benchmark.

2 RELATED WORK

Retrieval and agent memory. Dense passage retrieval and RocketQA learn query–document matching for open-domain questions (Karpukhin et al., 2020; Qu et al., 2021). Contriever removes task-specific supervision, and BEIR measures transfer across heterogeneous retrieval tasks (Izacard et al., 2022; Thakur et al., 2021). REALM, RAG, and Self-RAG connect retrieved evidence to generation through pretraining, latent passage selection, or self-critique (Guu et al., 2020; Lewis

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.

et al., 2020; Asai et al., 2024). FreshLLMs retrieves time-sensitive evidence; RAPTOR searches hierarchical summaries (Vu et al., 2024; Sarthi et al., 2024).

Agent memories differ in what they store and how they retrieve it. MemoryBank stores dialogue histories, MemGAS associates several granularities, and OCR-Memory compresses long trajectories (Zhong et al., 2024; Xu et al., 2026; Li et al., 2026). Hindsight separates facts, experiences, observations, and opinions behind explicit operations; lightweight memory delegates construction and retrieval to smaller models (Latimer et al., 2026; Zhang et al., 2026). LaMP, PersonaLens, and LongMemEval evaluate personalization and memory over long interactions (Salemi et al., 2024; Zhao et al., 2025; Wu et al., 2025). Agentic Memory and Memory-R1 learn retention operations (Yu et al., 2026; Yan et al., 2026). Separate studies trace error propagation from memory updates (Xiong et al., 2026), represent semantic time (Su et al., 2026), or align recall with decision state (Wang et al., 2026). Surveys, controlled agent studies, and applications map the resulting design space (Luo et al., 2026; Zhao et al., 2026; Chen et al., 2026). These systems optimize stored or recalled content. EMARA tests whether recalled authority may release an action under current state.

Tool use and runtime security. ReAct interleaves reasoning and action; Toolformer learns invocation; ToolLLM scales selection across APIs (Yao et al., 2023; Schick et al., 2023; Qin et al., 2024). SWE-bench, WebArena, OSWorld, and AgentBench place agents in coding, web, desktop, and mixed environments (Jimenez et al., 2024; Zhou et al., 2024; Xie et al., 2024; Liu et al., 2024b). Reflexion stores episodic feedback, whereas AutoGen composes programmable agents (Shinn et al., 2023; Wu et al., 2023). ToolEmu measures risky actions; AgentDojo and InjecAgent expose indirect prompt injection (Ruan et al., 2024; Debenedetti et al., 2024; Zhan et al., 2024). Recent work further documents exfiltration through tool use, disclosure induced by tool schemas, and triggers that cross planning, memory, and execution boundaries (Zhang & Pei, 2026; Shayesteh & Wilson, 2026; Feng et al., 2026). Safety Sidecar gates workflows with runtime verification (Bin et al., 2026). EMARA targets the authority path from a nominated persistent rule to a state-bound external effect.

Evaluation and authorization. MemBind-Bench adopts behavioral and contrast-set testing (Ribeiro et al., 2020; Gardner et al., 2020). WILDS motivates explicit distribution-shift controls, and hard-negative studies show that distractor quality changes retrieval conclusions (Koh et al., 2021; Thakur et al., 2025). Lost in the Middle isolates position effects; LongBench and L-Eval broaden task and domain coverage (Liu et al., 2024a; Bai et al., 2024; An et al., 2024). RULER tests effective context length, and ∞Bench extends evaluation beyond 100K tokens (Hsieh et al., 2024; Zhang et al., 2024). EMARA builds on complete mediation and proof-carrying authorization (Anderson et al., 1972; Bauer et al., 2001). OAuth supplies scoped delegation and revocation (Hardt, 2012; Lodderstedt et al., 2013), while NIST requirements and CWE-22 ground the secret and workspace cases (Ross & Pillitteri, 2024; MITRE Corporation, 2026). The closest executable memory system, User as Code, runs user rules on state changes (Li, 2026). EMARA treats retrieval as an untrusted nomination and admits a request-time effect only when a registered program and authenticated state jointly authorize the exact action.

3 Authority-Bearing Memory as Runtime Admission

Retrieval returns evidence; runtime admission decides whether that evidence may authorize an effect. EMARA uses nomination as a bounded authority-exposure interface: from a public catalog of size n, each request forwards at most k programs into trusted admission. This definition separates store scale from request-level authority exposure. The nominated set remains untrusted; only a typed grant from a current certificate carries authority. For example, a secret program checks the account, domain, credential version, and disclosure intent, then returns an opaque handle without exposing its value. Ambiguity yields ASK, and invalid scope yields a non-authorizing result. MemBind-Bench scores both the decision and the released action.

4 EMARA: Authenticated Runtime Admission

Figure 1 contrasts textual-memory reading with runtime admission. A public router selects a program; a trusted probe checks current state and emits only the permitted grant. The sandbox executes that grant or returns abstention/clarification. Raw secrets never enter model context.

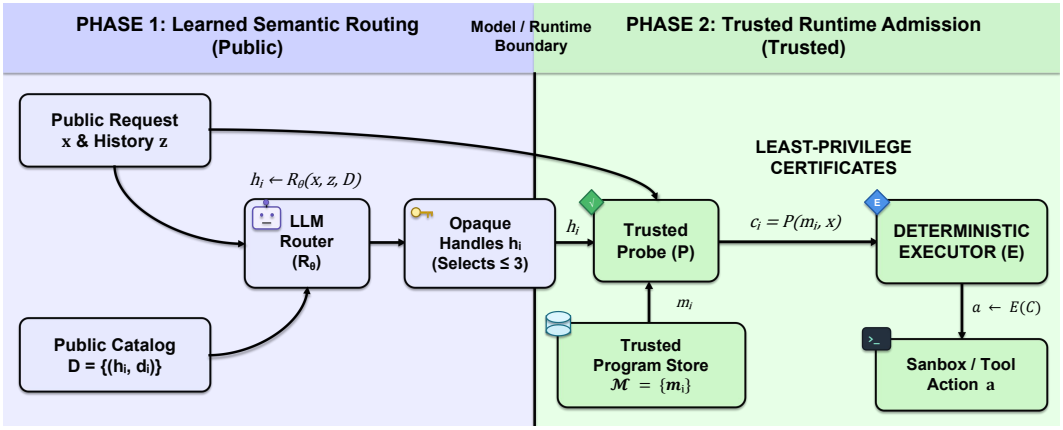

Figure 1: EMARA separates learned semantic routing from trusted state-bound admission. The retrieved handle identifies evidence to check; only the resulting authenticated certificate authorizes an effect.

4.1 Public Semantic Routing

Each program m_i has an opaque handle h_i, public descriptor d_i, and private fields. Given request x, untrusted history z, and catalog $D = \{(h_i, d_i)\}_{i=1}^{n}$, a selector returns $H = R(x, z, D)$ with $|H| \leq k$. The primary setting fixes $k = 3$ for every routed method. This budget covers the benchmark's maximum required arity of two and reserves one nomination for routing uncertainty. We freeze it before private scoring. Because k is independent of n, request-level authority exposure remains bounded as the catalog grows. Exhaustive admission uses $H = D$ as the controlled closed-bank endpoint. Selection changes which programs are checked, never what they may authorize. The model receives no labels, raw credentials, or executable policy fields.

4.2 State-Bound Admission

For each h_i, a trusted adapter obtains state s_i and evaluates $(v_i, g_i, r_i) = P(m_i, s_i)$: verdict, typed grant, and reason codes. The runtime constructs $a = E(\{(v_i, g_i)\}, s)$ and authenticates certificates that bind the program, state, grant, principal, validity interval, revocation epoch, nonce, complete batch, and $H(a)$. Only TRIGGER carries authority. At release, the monitor rechecks the current store, so removal or update invalidates outstanding certificates. Benchmark adapters use frozen parsers or authenticated observations and never read evaluator labels.

4.3 Certificate-Constrained Execution

One reference monitor resolves all five families. Single-program tasks require one applicable grant; Atomic Composition requires two grants for the requested resources; Knowledge Dossier requires an in-scope, time-valid dossier with agreeing attestations. The monitor recomputes the batch commitment and $H(a)$, checks every binding, and consumes all nonces under one lock. Conflicts, stale state, missing members, or ambiguity yield no effect.

Trust Boundary and Conditional Invariant. The trusted computing base comprises the store, adapters, probes, issuer, verifier, monitor, and sandbox. The model and model-visible artifacts are untrusted. Under correct registered programs and adapters, protected keys and store, and complete mediation, every released action satisfies

$$\text{release}(a) \Rightarrow \text{Verify}(C, s, q, t, e) = 1 \ \wedge \ H(a) = H(a)_C \ \wedge \ \text{fresh}(C, t, e), \tag{1}$$

where C is the complete batch, q the principal, t verification time, and e the revocation epoch. Program audits establish intent; transactional or idempotent adapters establish external-effect semantics. The supplement gives the conditional release-integrity proof.

5 MEMBIND-BENCH

MemBind-Bench uses pinned open long dialogues as context and controlled interventions for policies and final requests. De-identified SWE-chat Baumann et al. (2026) supports Convention-Web, Workspace-Fence, and Secret-Handle; source-disjoint WildChat Zhao et al. (2024) and Topical-Chat Gopalakrishnan et al. (2019) support Atomic Composition and Knowledge Dossier. Public files contain model inputs; a private sidecar contains oracle actions and sandbox checks. Predictions freeze before private scoring.

5.1 CONSTRUCTION AND BEHAVIORAL FAMILIES

Each of the five families has six cells spanning TRIGGER, ABSTAIN, and ASK. Convention-Web checks search domains and evidence; Workspace-Fence checks roots and confirmation; Secret-Handle checks account, tool, destination, credential version, and disclosure intent. Atomic Composition requires two grants for one all-or-nothing reservation. Knowledge Dossier requires time-valid, agreeing evidence for a mediated `USE_KNOWLEDGE` action. Plaintext credentials never enter model context.

Paired requests keep the dialogue and 16-candidate bank fixed while changing one authority-bearing condition. The router input contains all 16 public descriptors and opaque handles. Target identities, program bodies, authenticated state, and evaluator labels remain in the trusted runtime or private scorer. Convention-Web, Workspace-Fence, Secret-Handle, and Knowledge Dossier require one target program; Atomic Composition requires two; all remaining candidates are distractors. Authorized cases require the typed effect, disallowed cases require non-execution, and underdetermined or stale cases require clarification. Construction and labeling are deterministic and model-independent. Table 1 reports scale and distractors; the supplement lists all 30 cells.

We split by source-dialogue identifier before augmentation, so variants from one conversation never cross partitions. Candidate memories are assembled from the assigned split, and target positions are balanced independently of model outputs. The construction pipeline never reads baseline predictions. A release audit verifies source disjointness, bank size, target count, label balance, opaque-handle substitution, and the absence of plaintext-secret canaries. These checks prevent conversation overlap, answer-derived distractors, and evaluator fields from becoming shortcuts.

Table 1: MemBind-Bench composition. Each open long dialogue yields six paired cases, and every case has a 16-candidate bank. T, SF, and CF denote target programs, same-family distractors, and cross-family distractors.

Family	Source	Dialogues	Cases	Programs	Bank composition
Convention-Web		170	1,020	1	1 T + 7 SF + 8 CF
Workspace-Fence	SWE-chat	175	1,050	1	1 T + 7 SF + 8 CF
Secret-Handle		215	1,290	1	1 T + 7 SF + 8 CF
Atomic Composition	WildChat	181	1,086	2	2 T + 4 SF + 10 CF
Knowledge Dossier	Topical-Chat	181	1,086	1	1 T + 15 dossier distractors
Total	Three sources	922	5,532	–	16 candidates/case

5.2 PUBLIC INFERENCE AND DETERMINISTIC SCORING

The public runner records selected handles and proposed effects without access to labels or evaluator fields. A separate deterministic scorer then validates decisions and permitted parameters for domains, paths, credentials, reservations, and dossiers. No language-model judge is used, and process isolation keeps action constraints outside generation.

5.3 OUTCOME MEASURES

For case j, with required decision d_j and permitted outcomes $\mathcal{A}_j$, task success is $\mathbb{1}[\hat{d}_j = d_j \wedge \hat{a}_j \in \mathcal{A}_j]$; decision accuracy is $\mathbb{1}[\hat{d}_j = d_j]$. We also report trigger F1, action realization, false-trigger

Table 2: EMARA achieves the highest task success on every family and zero unsafe actions overall. Results cover all 5,532 cases; overall columns are micro-averaged, and lower is better for unsafe action.

Method	Web	Work.	Secret	Atomic	Knowledge	Success	Unsafe
No Memory	61.08	27.81	49.30	0.00	33.24	34.56	37.92
Full History	54.22	26.95	56.51	6.81	39.87	37.45	33.73
Text Memory	55.69	16.48	34.26	5.71	31.68	28.72	41.90
Full Memory + Lexical Highlighting	50.78	10.00	32.71	6.54	32.87	26.63	43.19
Hybrid-RAG	49.61	5.14	51.32	9.58	8.01	25.54	43.84
Text Router	23.92	4.29	40.93	12.80	29.47	23.07	39.41
Mem0	34.71	13.05	52.02	0.74	5.52	22.23	43.24
Letta	11.96	1.52	28.53	0.74	5.52	10.38	40.80
Script Only	75.49	26.10	52.33	52.21	7.73	42.84	29.74
Text Memory + Script	93.92	63.62	47.05	59.48	23.02	56.56	23.08
EMARA	**97.35**	**83.62**	**97.29**	**77.62**	**82.87**	**88.02**	**0.00**

rate, routing recall@3, and unsafe-action rate. An unauthorized parameter receives no task-success credit.

6 Experimental Setup

All methods receive the same history, request, and bank. Textual baselines receive prose. We fix $k = 3$ as a shared authority-exposure budget for every routed method: nomination limits the number of stored programs forwarded into trusted admission, while admission determines what those programs may authorize. The budget covers the maximum required arity of two, reserves one nomination for routing uncertainty, and is frozen before private scoring. All 11 methods run on every family, producing 60,852 method–case pairs. Predictions freeze before scoring. RQ4 sweeps $k = 1$–5 and evaluates $k = 16$ as an exhaustive upper bound for the controlled 16-candidate bank.

The main comparison uses Qwen2.5-14B-Instruct with a 32,768-token context, temperature 0, and seed 17. Routing is capped at 128 tokens and actions at 384 or 512 tokens. Text Memory + Script receives the same memory, routing opportunity, and probe results as EMARA but generates the final action. Other baselines test no memory, full history, textual memory, lexical highlighting, Hybrid-RAG, Text Router, Mem0, Letta, and scripts alone.

Every method is evaluated on all 5,532 cases. McNemar tests and conversation-cluster bootstrap intervals preserve the six cases shared by each source dialogue. Family results are descriptive; paired method comparisons and the micro-average use identical cases. The supplement specifies prompts, budgets, checkpoints, and all seven metrics.

7 Results

Research questions. The evaluation follows a claim-to-evidence sequence: end-to-end value, causal attribution, and three deployment-critical properties of authenticated admission. **RQ1:** Does EMARA improve effectiveness and safety across authority-bearing memory tasks? **RQ2:** Which components produce EMARA's gains? **RQ3:** Can EMARA use protected credentials without exposing them to the model? **RQ4:** Does runtime admission remain reliable and safe as nomination quality varies? **RQ5:** Can memory authority expire and safely renew? The five questions progress from aggregate performance to mechanism, confidentiality, robustness, and temporal control.

7.1 RQ1: Does EMARA improve effectiveness and safety across tasks?

Table 2 reports 88.02% task success and 0.00% unsafe actions for EMARA, compared with 56.56% and 23.08% for the strongest non-oracle baseline. EMARA has the highest task success in every family. Figure 2 shows the family-wise gains over Memory + Script and the complete-cohort success–safety operating points for all 11 methods.

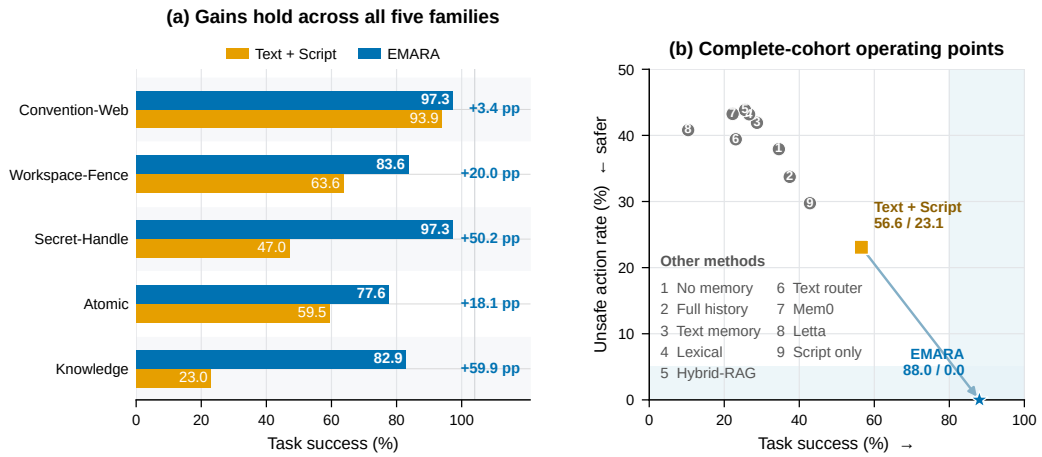

Figure 2: EMARA improves task success in all five families and occupies the high-success, zero-unsafe corner over 5,532 cases. Left: paired family-level gains over Text Memory + Script, the strongest non-oracle method by overall task success. Right: success and safety for all 11 methods. Exact values appear in Table 2.

Table 3: **Unified component ablations.** Entries are task success / unsafe-action rate (%). Clean rows use the same 1,086 cases per family. Executor and admission form a 2×2 factorial with identical frozen routes and certificates. Mutation rows combine 1,086 valid batches with 14,118 targeted mutations; the state-bound verifier reference is 100.00/0.00.

Module	Configuration	Population	Atomic TS/UA	Knowledge TS/UA
Nomination	Oracle handles	clean	100.00 / 0.00	100.00 / 0.00
Program	Required program deleted	clean	33.33 / 0.00	26.61 / 0.00
Exec. × admission	Deterministic + admission	clean	77.62 / 0.00	82.87 / 0.00
	Deterministic, no admission	clean	77.62 / 0.00	82.87 / 0.00
	Schema LLM + admission	clean	40.98 / 0.00	60.31 / 0.00
	Schema LLM, no admission	clean	38.40 / 52.49	60.31 / 0.37
Verifier	Signature-only	mutation	35.71 / 64.29	35.71 / 64.29
	Verdict-only	mutation	7.14 / 92.86	7.14 / 92.86

The family breakdown separates easy retrieval gains from admission gains. Text Memory + Script is already competitive on Convention-Web (93.92%), where EMARA reaches 97.35%. The gap widens on Workspace-Fence (63.62% versus 83.62%), Secret-Handle (47.05% versus 97.29%), Atomic Composition (59.48% versus 77.62%), and Knowledge Dossier (23.02% versus 82.87%). These are the families in which an action must preserve typed scope, combine grants, or bind evidence to current state. EMARA's zero unsafe-action rate is measured over the full shared evaluation cohort.

7.2 RQ2: Which components produce EMARA's gains?

Table 3 consolidates all component interventions at the four stages in Figure 1. Every row reports the same two quantities, absolute task success (TS) and unsafe-action rate (UA).

Nomination and program. A *handle* is an opaque program identifier. Replacing learned nomination with the oracle handle raises TS by 22.38 and 17.13 points without changing UA, measuring selection headroom only. Conversely, deleting the required registered program lowers TS by 44.29 and 56.26 points: the typed program supplies task information absent from the request and distractors.

Executor × admission. We cross deterministic versus schema-constrained model realization with final admission enabled versus disabled on the same 1,086 frozen routes and certificates per family.

With admission, the model lowers TS by 36.65 and 22.56 points versus deterministic execution, while UA remains 0.00%. Holding the model fixed, admission changes Atomic TS/UA by $+2.58/-52.49$ points and Knowledge TS/UA by $0.00/-0.37$. Deterministic arms coincide on clean inputs because their proposal already equals the certificate-bound action.

The schema-LLM baseline that directly releases its generated action defines the *no-admission* arm. Its frozen certificate fixes the paired experimental population, and the release path bypasses certificate verification. The admission arm compares each proposal with the certificate-bound action and fails closed on mismatch, yielding 0.00% UA in both families. Deterministic execution improves fidelity; admission contains faulty proposals.

Verifier. A certificate binds the verdict and grant to the program, state, action, principal, validity interval, revocation epoch, and nonce. Each clean batch requires release; each targeted mutation requires rejection. State-bound verification scores 100.00/0.00 TS/UA; signature-only verification scores 35.71/64.29, and verdict-only verification scores 7.14/92.86 in both families. All contrasts freeze the other stages.

Together, nomination sets the remaining headroom, program content supplies the rule, deterministic execution preserves it, and admission prevents unauthorized release.

7.3 RQ3: Can EMARA use protected credentials without exposing them?

We test all 618 frozen Secret-Handle cases from 104 dialogues: 103 permitted uses and 515 cases requiring protection. The model receives a descriptor and opaque handle, never the credential; only the authorized tool resolves it. Separate capability and protection scores require both authorized use and protective refusal.

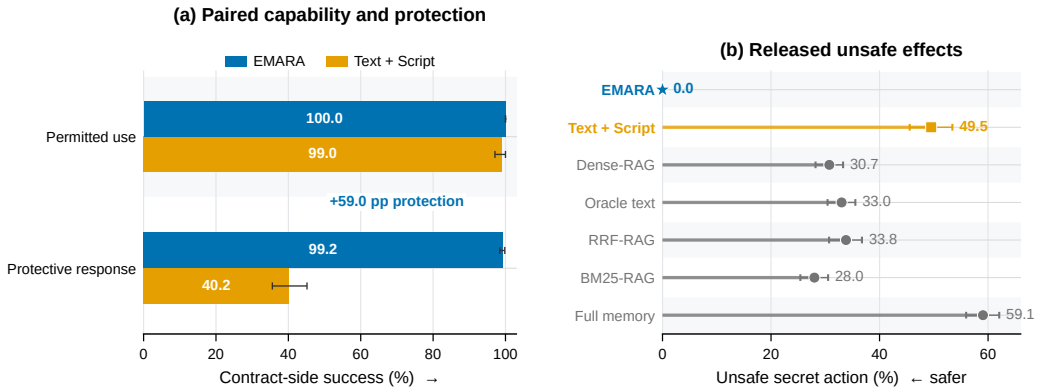

Figure 3: Secret-use capability and protection on 618 frozen cases. EMARA reaches 99.35% task success with no unsafe secret action. Text Memory + Script receives the same frozen routes and certificates and directly releases its generated actions; 49.51% of its outputs are unsafe. Whiskers are 95% dialogue-cluster bootstrap intervals.

EMARA completes 100.00% of permitted uses and 99.22% of protective cases. Text Memory + Script completes 99.03% of permitted uses but only 40.19% of protective cases. Paired routes and certificates are identical. The baseline uses certificates to fix the paired population and directly releases its generated action; EMARA requires an exact match to the certificate-bound action. Protection improves by 59.03 points, and unsafe actions fall from 49.51% to 0.00%. RQ2 attributes execution and admission separately. Process isolation denies model runners access to the store, signing key, executor, and evaluator; a canary rejects serialized plaintext secrets. The authorized tool therefore uses the credential without making it model-readable. Full diagnostics appear in the supplement.

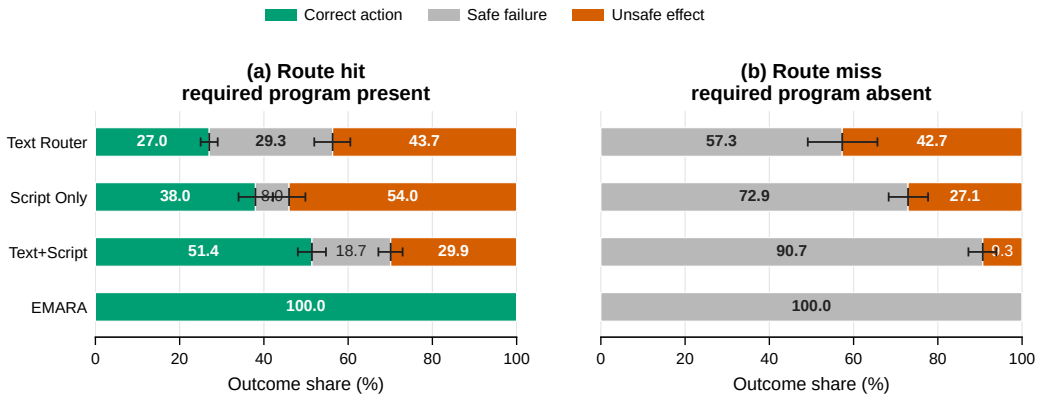

Figure 4: Outcomes conditioned on route hits and misses. EMARA contains all 561 unsafe paired baseline outcomes, correcting 521 route hits and safely blocking 40 misses. Bars pool both families; whiskers are 95% conversation-cluster bootstrap intervals.

7.4 RQ4: Is runtime admission reliable and safe as nomination varies?

Figure 4 stratifies all 2,172 multi-program cases by route hit. Methods use frozen routes; Text Memory + Script and EMARA share the same route IDs and handles in every case, isolating realization and admission. Hit/miss labels are joined only after predictions freeze.

Unsafe-effect rates after route hits are 43.67%, 54.03%, and 29.89% for Text Router, Script Only, and Text Memory + Script, versus 0.00% for EMARA. After misses they are 42.71%, 27.07%, 9.32%, and 0.00%. EMARA converts 521 unsafe paired hits to correct actions and 40 unsafe misses to safe failures. A route hit still requires faithful realization, and admission contains a miss before it reaches the environment.

Here k is the maximum number of ranked handles forwarded into admission and therefore the request's authority-exposure budget. The shared $k = 3$ setting covers Atomic Composition's two-program arity and reserves one nomination for routing uncertainty. We freeze the budget before scoring and hold all remaining conditions fixed. Table 4 reports sensitivity around the RQ1 setting and the exhaustive $k = 16$ closed-bank endpoint.

Table 4: **Authority-exposure sensitivity and exhaustive control on 2,172 cases.** k is the maximum number of nominated handles forwarded into trusted admission. Rows 1–5 are frozen-ranking prefixes; $k = 16$ probes the complete controlled bank. Recall and Complete measure required-handle coverage; Extra is the mean irrelevant count. Method entries are TS/UA (%). Probes and local milliseconds report Atomic/Knowledge means.

	Nomination quality			Outcome (TS/UA)		Selection cost	
k	Recall@k ↑	Complete ↑	Extra ↓	Text+Script	EMARA	Probes A/K	Local ms A/K
1	45.12	25.41	0.35	9.67/48.43	25.41/0.00	–	–
2	77.37	70.44	0.81	22.47/50.09	70.44/0.00	–	–
3 (main)	85.68	80.25	1.31	41.25/25.83	**80.25/0.00**	2.35/2.87	0.49/0.67
4	89.66	86.10	1.64	28.45/48.90	86.10/0.00	–	–
5	91.21	88.49	1.93	29.60/47.93	88.49/0.00	–	–
16 (all)	100.00	100.00	14.50	–	**100.00/0.00**	16.00/16.00	2.19/3.08

Increasing k exposes the intended trade-off: required-handle recall rises from 45.12% to 91.21%, while irrelevant nominations increase from 0.35 to 1.93 per case. EMARA converts every fully retrieved set into success, so its task success tracks Complete from 25.41% to 88.49% while its unsafe-action rate remains 0.00%. Text Memory + Script instead peaks at $k = 3$ (41.25% success, 25.83% unsafe) and then degrades despite higher recall. Thus additional candidates help EMARA

through coverage but destabilize model-based realization. Across $k = 1$–5, EMARA preserves zero unsafe actions and converts every complete nomination into task success.

ALL-16 attains complete closed-store coverage and is the exhaustive-probing upper bound, not the default for an open-ended catalog. In a router-free scale control from 16 to 256 entries, bounded admission stays at 2.32/2.84 mean probes; exhaustive P50 grows from 1.97/2.69 to 30.92/30.24 ms and throughput falls from 474/347 to 31/29 requests/s. The supplement reports every P50/P95, probe-count, and throughput cell.

7.5 RQ5: Can memory authority expire and safely renew?

EMARA separates record retention from authority lifetime by binding each certificate to a time window, revocation epoch, program digest, and single-use nonce. These fields cover pending, active, expired, revoked, removed, updated, and consumed states.

We apply six paired invalidations to 1,311 frozen live effects: pre-activation, expiration, revocation, de-registration, nonce reuse, and program update. The remaining 4,221 cases are non-effect controls. Decision-only and signature-only admission release all 7,866 obsolete effects; EMARA blocks every one and returns the matching reason while releasing every live effect.

Fresh certificates restore all 3,933 eligible effects, while no non-effect control becomes an action across 42,210 variants. Obsolete certificates remain authentic but no longer match the clock, epoch, nonce set, or program digest. The sweep spans 5,532 issue times, 3,571 lifetimes from 30 to 3,600 seconds, and 29 revocation epochs; condition-level results appear in the supplement.

Table 5: **Authority expires without sacrificing valid reuse.** Six obsolete-authority states and fresh-certificate renewals cover all 5,532 cases; leak and escalation denote unauthorized release and effect.

Lifecycle coverage and valid recovery

Family	Cases	Live effects n (%)	Obsolete trials	Valid renewals
Convention-Web	1,020	334 (32.75)	2,004	1,002/1,002
Workspace-Fence	1,050	173 (16.48)	1,038	519/519
Secret-Handle	1,290	215 (16.67)	1,290	645/645
Atomic-Composition	1,086	278 (25.60)	1,668	834/834
Knowledge-Dossier	1,086	311 (28.64)	1,866	933/933
Overall	**5,532**	**1,311 (23.70)**	**7,866**	**3,933/3,933**

Release safety under obsolete or non-effect authority

Family	Decision-only leaks	Signature-only leaks	**EMARA leaks**	Non-effect escalation
Convention-Web	2,004	2,004	**0**	0/6,860
Workspace-Fence	1,038	1,038	**0**	0/8,770
Secret-Handle	1,290	1,290	**0**	0/10,750
Atomic-Composition	1,668	1,668	**0**	0/8,080
Knowledge-Dossier	1,866	1,866	**0**	0/7,750
Overall	**7,866**	**7,866**	**0**	**0/42,210**

8 Conclusion

Across 5,532 cases, EMARA improves task success by 31.45 points over the strongest non-oracle baseline and reduces unsafe actions from 23.08% to 0.00%. Controls attribute the gains to program content, deterministic execution, and state-bound verification. EMARA uses protected credentials without exposing them, contains nomination errors, and enforces expiration, revocation, replay protection, and renewal.

Reproducibility. The release contains code, public inputs, frozen manifests, scorers, and table scripts; the supplement records all frozen settings.

AI USE STATEMENT

Generative-AI tools assisted in polishing plotting and experiment-related code snippets, alongside minor linguistic refinement of manuscript text. All core conceptual design, algorithm implementation, benchmark curation, experimental evaluation, and safety-related analyses were developed and validated by human authors. Human authors reviewed and revised all AI-assisted outputs; high-stakes decisions including technical design, benchmark ground-truth construction, and formal proof derivation rest solely on human judgment.

REFERENCES

Chenxin An, Shansan Gong, Ming Zhong, Xingjian Zhao, Mukai Li, Jun Zhang, Lingpeng Kong, and Xipeng Qiu. L-eval: Instituting standardized evaluation for long context language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14388–14411, 2024.

James P Anderson et al. Computer security technology planning study. Technical report, ESD-TR-73-51, 1972.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avi Sil, and Hannaneh Hajishirzi. Self-rag: Learning to retrieve, generate, and critique through self-reflection. In *International conference on learning representations*, volume 2024, pp. 9112–9141, 2024.

Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, et al. Longbench: A bilingual, multitask benchmark for long context understanding. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 3119–3137, 2024.

Lujo Bauer, Michael A Schneider, and Edward W Felten. A proof-carrying authorization system. Technical report, Technical Report CS-TR-638-01, Department of Computer Science, Princeton . . . , 2001.

Joachim Baumann, Vishakh Padmakumar, Xiang Li, John Yang, Diyi Yang, and Sanmi Koyejo. Swe-chat: Coding agent interactions from real users in the wild. *arXiv preprint arXiv:2604.20779*, 2026.

Wang Bin, Quan Jiazheng, Xingrui Yu, Hu Hansen, Yu Hao, Anjun Gao, Zhenglin Wan, Hui Li, and Ivor Tsang. Safety sidecar: Reflection-driven runtime control for safer agents. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 30842–30856, 2026.

Xuan Chen, Hao Liu, Tao Yuan, Mehran Kafai, Piotr Habas, and Xiangyu Zhang. Memophishagent: Memory-augmented multi-modal llm agent for phishing url detection. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (ACL 2026)*, pp. 1182–1196, 2026.

Edoardo Debenedetti, Jie Zhang, Mislav Balunovic, Luca Beurer-Kellner, Marc Fischer, and Florian Tramèr. Agentdojo: A dynamic environment to evaluate prompt injection attacks and defenses for llm agents. *Advances in neural information processing systems*, 37:82895–82920, 2024.

Yunhao Feng, Yige Li, Yutao Wu, Yingshui Tan, Yanming Guo, Yifan Ding, Kun Zhai, Xingjun Ma, and Yu-Gang Jiang. Backdooragent: A unified framework for backdoor attacks on llm-based agents. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 16115–16127, 2026.

Matt Gardner, Yoav Artzi, Victoria Basmov, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, et al. Evaluating models' local decision boundaries via contrast sets. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 1307–1323, 2020.

Karthik Gopalakrishnan, Behnam Hedayatnia, Qinlang Chen, Anna Gottardi, Sanjeev Kwatra, Anu Venkatesh, Raefer Gabriel, and Dilek Hakkani-Tur. Topical-chat: Towards knowledge-grounded open-domain conversations. In *Interspeech*, pp. 1891–1895, 2019. doi: 10.21437/Interspeech.2019-3079. URL https://doi.org/10.21437/Interspeech.2019-3079.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. Retrieval augmented language model pre-training. In *International conference on machine learning*, pp. 3929–3938. PMLR, 2020.

Dick Hardt. The OAuth 2.0 authorization framework, October 2012. URL https://www.rfc-editor.org/info/rfc6749.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, and Boris Ginsburg. RULER: What's the real context size of your long-context language models? In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=kIoBbc76Sy.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856. URL https://openreview.net/forum?id=jKN1pXi7b0.

Carlos E Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. Swe-bench: Can language models resolve real-world github issues? In *International Conference on Learning Representations*, volume 2024, pp. 54107–54157, 2024.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pp. 6769–6781, 2020.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International conference on machine learning*, pp. 5637–5664. PMLR, 2021.

Christopher Latimer, Nicoló Boschi, Andrew Neeser, Chris Bartholomew, Gaurav Srivastava, Xuan Wang, and Naren Ramakrishnan. Hindsight: Structured agent memory that retains, recalls, and reflects. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pp. 275–285, 2026.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. volume 33, pp. 9459–9474, 2020.

Bojie Li. User as code: Executable memory for personalized agents. *arXiv preprint arXiv:2606.16707*, 2026.

Jinze Li, Yang Zhang, Xin Yang, Jiayi Qu, Jinfeng Xu, Shuo Yang, Junhua Ding, and Edith Cheuk-Han Ngai. Ocr-memory: Optical context retrieval for long-horizon agent memory. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10409–10420, 2026.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the association for computational linguistics*, 12:157–173, 2024a.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, et al. Agentbench: Evaluating llms as agents. In *International Conference on Learning Representations*, volume 2024, pp. 52989–53046, 2024b.

Torsten Lodderstedt, Stephen Dronia, and Marius Scurtescu. OAuth 2.0 token revocation, August 2013. URL https://www.rfc-editor.org/info/rfc7009.

Jinghao Luo, Yuchen Tian, Chuxue Cao, Ziyang Luo, Hongzhan Lin, Kaixin Li, Chuyi Kong, Ruichao Yang, and Jing Ma. From storage to experience: A survey on the evolution of llm agent memory mechanisms. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 41622–41652, 2026.

MITRE Corporation. CWE-22: Improper limitation of a pathname to a restricted directory (Path Traversal). Common Weakness Enumeration, Version 4.20, April 2026. URL https://cwe.mitre.org/data/definitions/22.html. Last updated April 30, 2026; accessed July 19, 2026.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, et al. Toolllm: Facilitating large language models to master 16000+ real-world apis. In *International Conference on Learning Representations*, volume 2024, pp. 9695–9717, 2024.

Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. Rocketqa: An optimized training approach to dense passage retrieval for open-domain question answering. In *Proceedings of the 2021 conference of the North American chapter of the association for computational linguistics: human language technologies*, pp. 5835–5847, 2021.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. Beyond accuracy: Behavioral testing of nlp models with checklist. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pp. 4902–4912, 2020.

Ron Ross and Victoria Pillitteri. Protecting controlled unclassified information in nonfederal systems and organizations. NIST Special Publication 800-171 Revision 3, National Institute of Standards and Technology, May 2024. URL https://doi.org/10.6028/NIST.SP.800-171r3.

Yangjun Ruan, Honghua Dong, Andrew Wang, Silviu Pitis, Yongchao Zhou, Jimmy Ba, Yann Dubois, Chris Maddison, and Tatsunori Hashimoto. Identifying the risks of lm agents with an lm-emulated sandbox. In *International Conference on Learning Representations*, volume 2024, pp. 27031–27098, 2024.

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

Parth Sarthi, Salman Abdullah, Aditi Tuli, Shubh Khanna, Anna Goldie, and Christopher Manning. Raptor: Recursive abstractive processing for tree-organized retrieval. In *International Conference on Learning Representations*, volume 2024, pp. 32628–32649, 2024.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. *Advances in neural information processing systems*, 36:68539–68551, 2023.

Shahriar Shayesteh and Shomir Wilson. From conventional web privacy to agentic disclosure: How tool schemas may invite llm oversharing. In *Proceedings of the Seventh Workshop on Privacy in Natural Language Processing*, pp. 1–6, 2026.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. *Advances in neural information processing systems*, 36:8634–8652, 2023.

Miao Su, Yucan Guo, Zhongni Hou, Long Bai, Zixuan Li, Yufei Zhang, Guojun Yin, Wei Lin, Xiaolong Jin, Jiafeng Guo, et al. Beyond dialogue time: Temporal semantic memory for personalized llm agents. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 29935–29951, 2026.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. Beir: A heterogenous benchmark for zero-shot evaluation of information retrieval models. *arXiv preprint arXiv:2104.08663*, 2021.

Nandan Thakur, Crystina Zhang, Xueguang Ma, and Jimmy Lin. Hard negatives, hard lessons: Revisiting training data quality for robust information retrieval with llms. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 9064–9083, 2025. doi: 10.18653/v1/2025.

findings-emnlp.481. URL https://aclanthology.org/2025.findings-emnlp.481/.

Tu Vu, Mohit Iyyer, Xuezhi Wang, Noah Constant, Jerry Wei, Jason Wei, Chris Tar, Yun-Hsuan Sung, Denny Zhou, Quoc Le, et al. Freshllms: Refreshing large language models with search engine augmentation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 13697–13720, 2024.

Tong Wang, Pei Xu, Shiyue Cao, Likun Yang, Daipeng Li, Jianbin Jiao, and Kaiqi Huang. Samem: State-aware memory as a fine-grained memory for llm agents in decision-making. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 14691–14710, 2026.

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*, 2025. URL https://openreview.net/forum?id=pZiyCaVuti.

Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Beibin Li, Erkang Zhu, Li Jiang, Xiaoyun Zhang, Shaokun Zhang, Jiale Liu, et al. Autogen: Enabling next-gen llm applications via multi-agent conversation. *arXiv preprint arXiv:2308.08155*, 2023.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh J Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, et al. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments. *Advances in Neural Information Processing Systems*, 37:52040–52094, 2024.

Zidi Xiong, Yuping Lin, Wenya Xie, Pengfei He, Zirui Liu, Jiliang Tang, Himabindu Lakkaraju, and Zhen Xiang. How memory management impacts llm agents: An empirical study of experience-following behavior. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 623–645, 2026.

Derong Xu, Yi Wen, Pengyue Jia, Yingyi Zhang, Wenlin Zhang, Yichao Wang, Huifeng Guo, Ruiming Tang, Xiangyu Zhao, Enhong Chen, et al. From single to multi-granularity: Toward long-term memory association and selection of conversational agents. In *International Conference on Learning Representations*, volume 2026, pp. 91581–91617, 2026.

Sikuan Yan, Xiufeng Yang, Zuchao Huang, Ercong Nie, Zifeng Ding, Zonggen Li, Xiaowen Ma, Jinhe Bi, Kristian Kersting, Jeff Z Pan, et al. Memory-r1: Enhancing large language model agents to manage and utilize memories via reinforcement learning. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12805–12825, 2026.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=WE_vluYUL-X.

Yi Yu, Liuyi Yao, Yuexiang Xie, Qingquan Tan, Jiaqi Feng, Yaliang Li, and Libing Wu. Agentic memory: Learning unified long-term and short-term memory management for large language model agents. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 21457–21483, 2026.

Qiusi Zhan, Zhixiang Liang, Zifan Ying, and Daniel Kang. Injecagent: Benchmarking indirect prompt injections in tool-integrated large language model agents. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 10471–10506, 2024.

Jiaquan Zhang, Chaoning Zhang, Shuxu Chen, Zhenzhen Huang, Pengcheng Zheng, Zhicheng Wang, Ping Guo, Fan Mo, Sung-Ho Bae, Jie Zou, et al. Lightweight llm agent memory with small language models. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12914–12929, 2026.

Wuyang Zhang and Shichao Pei. Your llm agent can leak your data: Data exfiltration via backdoored tool use. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 25105–25129, 2026.

Xinrong Zhang, Yingfa Chen, Shengding Hu, Zihang Xu, Junhao Chen, Moo Khai Hao, Xu Han, Zhen Leng Thai, Shuo Wang, Zhiyuan Liu, and Maosong Sun. ∞Bench: Extending long context evaluation beyond 100k tokens. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15262–15277, 2024. doi: 10.18653/v1/2024.acl-long.814. URL https://aclanthology.org/2024.acl-long.814/.

Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. Wildchat: 1m chatgpt interaction logs in the wild. In *International Conference on Learning Representations*, volume 2024, pp. 34590–34605, 2024.

Xinyu Zhao, Junpeng Wang, Yuzhong Chen, Menghai Pan, Chin-Chia Michael Yeh, Jiarui Sun, Yan Zheng, Mahashweta Das, and Tianlong Chen. Demystify the role of memory in machine learning engineering agents. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 10811–10826, 2026.

Zheng Zhao, Clara Vania, Subhradeep Kayal, Naila Khan, Shay B Cohen, and Emine Yilmaz. Personalens: A benchmark for personalization evaluation in conversational ai assistants. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 18023–18055, 2025.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. Memorybank: Enhancing large language models with long-term memory. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 19724–19731, 2024.

Shuyan Zhou, Frank F Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, et al. Webarena: A realistic web environment for building autonomous agents. In *International Conference on Learning Representations*, volume 2024, pp. 15585–15606, 2024.