RECOVERMEM: RISK-CONTROLLED TRUST IN COMPRESSED AGENT MEMORY

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-horizon LLM agents compress growing interaction histories into long-term memory, but always trusting compressed memory risks acting on insufficient evidence, while always recovering additional evidence forgoes selective memory use. We introduce ReCoverMem, a TRUST/RECOVER controller that ranks decisions with a learned memory-sufficiency score and applies unit-level conformal risk control. Our audit shows that treating dependent decisions as independent examples weakens the finite-sample correction. Across three workloads and four downstream backbones, always trusting memory yields false-safe risk of 45–72%. At $\alpha = .10$, ReCoverMem routes 10–22% of decisions to TRUST, achieving lower false-safe risk and lower exceedance in all 12 configurations. When recovery is substantially more costly, selective trust also reduces execution cost: on NarrativeXL, ReCoverMem saves 8.5–13.6% of executed-branch tokens at $\alpha = .10$ and 3.9% at $\alpha = .05$.

1 INTRODUCTION

Long-horizon LLM agents use long-term memory to retain information as interactions accumulate (Packer et al., 2023; Zhong et al., 2024; Chhikara et al., 2025). By compressing past interactions into retrievable records, memory avoids repeatedly processing the full history. Yet memory sufficiency is decision-dependent: a record may preserve a general preference while omitting an exception that determines the next action. The same retrieved memory can therefore support some decisions and fail others.

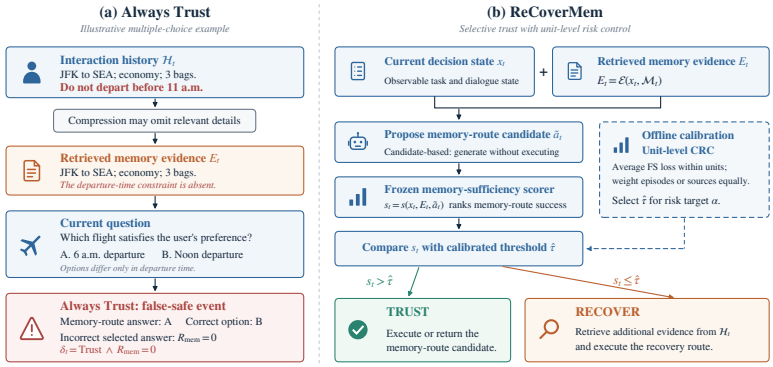


Figure 1: Compressed memory creates a selective trust problem. **(a)** In this illustrative example, compression omits the user's departure-time constraint. ALWAYS TRUST returns an incorrect memory-route answer, producing a false-safe (TRUST with $R_{\text{mem}} = 0$). **(b)** In the candidate-based workflow shown, ReCoverMem proposes a memory-route candidate without executing it and scores memory sufficiency using a frozen scorer conditioned on the current state, retrieved evidence, and candidate. A threshold calibrated by unit-level CRC routes execution to TRUST when $s_t > \hat{\tau}$ and to RECOVER otherwise, without modifying the host memory. False-safes are evaluated from matched checkpointed states (Section 4.1).

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.

Recent work has examined failure modes in long-term memory and their effects on downstream agent decisions (Deng et al., 2026; Hao et al., 2026; Long et al., 2026; Ma et al., 2026). This motivates a complementary execution-time question: *when should an agent trust compressed memory?* Always trusting risks acting on insufficient evidence, while always recovering additional historical records fails to exploit cases where compressed memory is already sufficient for the current decision. An agent therefore needs a decision-specific mechanism for choosing whether to act from compressed memory or recover additional evidence, with an explicit budget for the risk of misplaced trust.

Turning this choice into a risk-controlled policy raises two statistical challenges. First, a learned memory-sufficiency score can rank decisions, but ranking alone does not control the risk of trusting memory. Conformal risk control (CRC) provides a way to calibrate a routing threshold to a user-specified risk level under exchangeability (Angelopoulos et al., 2024). Second, agent-memory decisions are naturally grouped: checkpoints from the same episode share an interaction history, while queries from the same source may depend on overlapping evidence. Calibration must therefore operate over appropriate episode- or source-level units rather than treating all decisions as independent examples (Lee et al., 2023).

We introduce ReCoverMem, a risk-controlled TRUST/RECOVER controller for compressed agent memory. A *false-safe* (FS) occurs when the controller chooses TRUST but the memory route fails a predefined operational success criterion. ReCoverMem learns a memory-sufficiency score to rank decisions and uses CRC over episode- or source-level units to select a routing threshold for a target FS risk budget. For a fixed scorer and decision-generation protocol, CRC controls expected unit-averaged FS loss when calibration units and future episodes or sources are exchangeable. This separates the scorer's ranking role from the statistical role of calibration.

Our analysis distinguishes the effects of calibration-unit counts and loss weighting, and characterizes how memory-sufficiency prevalence constrains the available risk–coverage region. Across workloads, downstream backbones, and host memories, we evaluate how scoring, calibration, and candidate generation affect risk, coverage, and routing cost. Paired fixed-state and on-policy evaluations distinguish memory sufficiency, recovery effectiveness, and final task success, while long-history experiments quantify token savings and their associated success trade-offs.

2 RELATED WORK

Agent memory and sufficiency routing. Several memory systems adapt memory retrieval at inference time. ReadAgent (Lee et al., 2024) combines gist memories with selective lookup of original passages. TierMem (Zhu et al., 2026) uses a sufficiency router to escalate from compact summaries to provenance-linked raw evidence. Jev-Mem (Jiang et al., 2026) uses lightweight control for memory construction, adaptive retrieval, and evidence-based stopping. These methods determine when additional memory content should be retrieved and when retrieval should stop. ReCoverMem differs in how the operating point is selected: we calibrate the TRUST/RECOVER threshold to an explicit false-safe risk target and evaluate the resulting trust coverage.

Selective prediction and operational sufficiency. Selective prediction trades coverage for reliability through rejection (Chow, 1970; Geifman & El-Yaniv, 2017; 2019), while learning to defer routes difficult cases to another decision-maker (Madras et al., 2018). LLM work uses confidence, self-evaluation, and semantic uncertainty to estimate correctness (Kadavath et al., 2022; Kuhn et al., 2023). Sufficient Context (Joren et al., 2025) separates evidence sufficiency from successful evidence use. We define memory sufficiency operationally by memory-route success and control the marginal risk of trusting when that route fails.

Conformal risk control. Conformal prediction gives marginal guarantees under exchangeability (Vovk et al., 2005; Angelopoulos & Bates, 2023), and conformal risk control extends calibration to bounded monotone losses (Angelopoulos et al., 2024). Related LLM applications include conformal factuality, language modeling, and abstention (Mohri & Hashimoto, 2024; Quach et al., 2024; Abbasi Yadkori et al., 2024). Adaptive retrieval (Jiang et al., 2023; Asai et al., 2024; Jeong et al., 2024) and LLM cascades (Chen et al., 2024; Ong et al., 2025) decide when evidence or a model suffices, without finite-sample error bounds. Agent-focused work includes reliability-constrained orchestration (Si et al., 2026) and SafeCommit (Akewar & Ranjan, 2026), which certifies side-effectful actions against a calibrated set of plausible worlds under memory uncertainty. In contrast, ReCoverMem calibrates trust in compressed memory under the operational success criterion.

3 METHOD

3.1 MEMORY TRUST AND FALSE-SAFE RISK

At a controlled decision state x_t, let $\mathcal{H}_t$ denote the interaction history and $\mathcal{M}_t$ the host system's long-term memory. The host retrieves

$$E_t = \mathcal{E}(x_t, \mathcal{M}_t).$$

ReCoverMem decides whether to TRUST this memory route or RECOVER additional evidence from $\mathcal{H}_t$, without modifying the host memory system.

Operational memory sufficiency. Let $R_{\mathrm{mem},t} \in \{0, 1\}$ indicate whether the memory route satisfies a predefined operational success criterion under a frozen backbone and decoding protocol. This defines decision-specific memory sufficiency without attributing failure to storage, retrieval, or downstream use. For diagnostics, we also evaluate a frozen recovery route from the same checkpoint, yielding $R_{\mathrm{rec},t}$. Neither outcome is available at routing time.

False-safe risk and trust coverage. A *false-safe* occurs when the controller trusts an operationally insufficient memory route:

$$\mathrm{FS}_\delta(i, t) = \mathbb{1}[\delta_{i,t} = \textsc{Trust} \wedge R_{\mathrm{mem},i,t} = 0] .$$

The statistical unit is an episode or source. For a non-empty unit with controlled decisions $\mathcal{T}_i$, define its false-safe loss and trust coverage as

$$\ell_i(\delta) = \frac{1}{|\mathcal{T}_i|} \sum_{t \in \mathcal{T}_i} \mathrm{FS}_\delta(i, t),$$

$$c_i(\delta) = \frac{1}{|\mathcal{T}_i|} \sum_{t \in \mathcal{T}_i} \mathbb{1}[\delta_{i,t} = \textsc{Trust}] .$$

Units receive equal weight; retained empty units contribute zero. The empty-unit convention is fixed before calibration and shared across calibration and evaluation.

For a fixed policy,

$$\mathrm{FS}(\delta) = \mathbb{E}[\ell_i(\delta)], \qquad \mathrm{Cov}(\delta) = \mathbb{E}[c_i(\delta)].$$

Our objective is

$$\max_{\delta} \ \mathrm{Cov}(\delta) \qquad \text{s.t.} \qquad \mathrm{FS}(\delta) \le \alpha.$$

3.2 SUFFICIENCY SCORING AND CONFORMAL CALIBRATION

Memory-sufficiency scoring. In the primary candidate-based formulation, the memory route proposes a candidate action $\tilde{a}_t$ without executing it, and ReCoverMem assigns

$$s_t = s(x_t, E_t, \tilde{a}_t).$$

Our primary scorer is logistic regression,

$$s_t = \sigma\big(w^\top \phi(x_t, E_t, \tilde{a}_t) + b\big) ,$$

using features of evidence coverage, retrieval similarity, candidate support and confidence, and decision progress. Feature selection, preprocessing parameters, and scorer fitting use only predictor-training data and are frozen before calibration.

We also evaluate a candidate-free variant,

$$s_t^{\mathrm{cf}} = s_{\mathrm{cf}}(x_t, E_t),$$

which excludes candidate-dependent features and permits routing before answer generation. In either case, the frozen score ranks decisions while CRC selects the operating threshold. Implementation details and auxiliary scorer analyses are provided in Appendix C.1.

Routing and calibration. For threshold τ,

$$\delta_\tau(t) = \begin{cases} \textsc{Trust}, & s_t > \tau, \\ \textsc{Recover}, & s_t \le \tau. \end{cases}$$

Ties therefore recover. For the candidate-free variant, s_t is replaced by s_t^{cf}.

Decision sets, finite scores, and memory-route outcomes are fixed independently of the threshold. For each non-empty calibration unit,

$$L_i(\tau) = \frac{1}{|\mathcal{T}_i|} \sum_{t \in \mathcal{T}_i} \mathbb{1}[s_{i,t} > \tau \wedge R_{\mathrm{mem},i,t} = 0],$$

with $L_i(\tau) = 0$ for retained empty units. These losses are bounded in $[0, 1]$, non-increasing, and right-continuous in τ.

For $n \geq 1$ calibration units, let

$$\widehat{L}_n(\tau) = \frac{1}{n} \sum_{i=1}^{n} L_i(\tau).$$

Let $\mathcal{C}$ contain the distinct calibration scores and $\{-\infty, +\infty\}$. Following conformal risk control (Angelopoulos et al., 2024), we select

$$\hat{\tau} = \begin{cases} \min\left\{\tau \in \mathcal{C} : \dfrac{n}{n+1}\widehat{L}_n(\tau) + \dfrac{1}{n+1} \leq \alpha\right\}, & \text{if non-empty,} \\ +\infty, & \text{otherwise.} \end{cases}$$

The smallest feasible threshold maximizes coverage among candidates satisfying the calibration criterion; the $+\infty$ fallback always recovers and incurs zero false-safe loss. Algorithm 1 in the appendix summarizes unit-level calibration and decision-time routing.

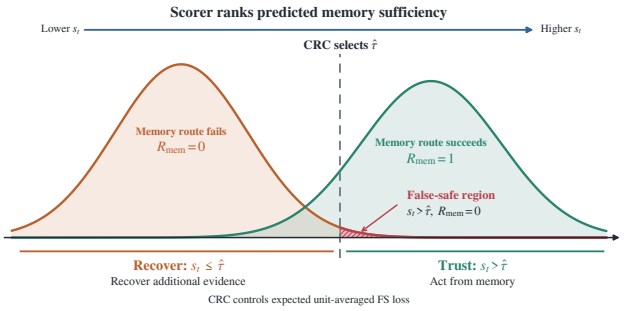


Figure 2: **Memory-sufficiency scoring and unit-level calibration.** The scorer ranks decisions, while CRC selects the strict threshold $\hat{\tau}$. Decisions with $s_t > \hat{\tau}$ are trusted; all others recover. A false-safe occurs when an insufficient memory route is trusted. Score distributions are schematic.

Risk guarantee. Conditional on the frozen scorer and decision-generation protocol, assume the calibration-unit loss functions and a future unit's loss function are exchangeable. Then

$$\mathbb{E}[L_{\mathrm{test}}(\hat{\tau})] \leq \alpha.$$

The expectation is taken over both the calibration sample and a future exchangeable unit, so the guarantee does not require every realized test set to have empirical FS below α. It controls marginal false-safe risk rather than the error rate conditional on trusting. Unit-level aggregation allows dependence within a unit, but the guarantee still relies on exchangeability across units and does not automatically extend to distribution shifts or routing-induced on-policy trajectories.

3.3 Risk–Coverage Geometry

For a non-empty unit, define its memory-sufficient fraction

$$p_i = \frac{1}{|\mathcal{T}_i|} \sum_{t \in \mathcal{T}_i} R_{\mathrm{mem},i,t},$$

with $p_i = 0$ for retained empty units. Let

$$\pi = \mathbb{E}[p_i], \qquad \rho = \Pr(|\mathcal{T}_i| > 0).$$

For any fixed routing policy,

$$c_i(\delta) \le p_i + \ell_i(\delta), \qquad c_i(\delta) \le \mathbb{1}[|\mathcal{T}_i| > 0].$$

Thus

$$\mathrm{Cov}(\delta) \le \min\{\rho, \pi + \mathrm{FS}(\delta)\}.$$

Any policy satisfying $\mathrm{FS}(\delta) \le \alpha$ therefore obeys

$$\mathrm{Cov}(\delta) \le \min\{\rho, \pi + \alpha\}.$$

For populations restricted to non-empty units, $\rho = 1$. Under the CRC false-safe guarantee, the corresponding expected trust coverage is bounded by

$$\mathbb{E}[c_{\mathrm{test}}(\delta_{\hat{\tau}})] \le \min\{\rho, \pi + \alpha\}.$$

Thus, when $\pi + \alpha$ is small, attainable trust coverage is necessarily limited. At the other extreme, ALWAYS TRUST is risk-feasible when $\pi \ge \rho - \alpha$. Between these regimes, ranking quality and calibration determine the coverage retained.

3.4 COST–COVERAGE CONDITION

Selective trust reduces cost only when the trusted branch is cheaper than recovery. Treating per-decision branch costs as constant, let $r = c_{\mathrm{trust}}/c_{\mathrm{rec}}$ denote the branch cost ratio, and let $m = \min\{\rho, \pi + \alpha\}$ be the upper bound on trust coverage for any risk-feasible policy from Section 3.3. We treat the lightweight scoring computation itself as negligible.

For candidate-free routing, the controller scores before answer generation, so only the selected branch is executed:

$$\frac{C(\delta)}{C_{\mathrm{rec}}} = \mathrm{Cov}(\delta)\, r + \big(1 - \mathrm{Cov}(\delta)\big). \tag{1}$$

When $r < 1$, a risk-feasible policy can therefore reduce normalized cost by at most $m(1 - r)$, regardless of ranking quality. This ceiling vanishes as $r \to 1$.

Candidate-based routing instead generates the memory-route candidate before routing and reuses it on TRUST. Every decision therefore pays the memory-route cost, while recovered decisions pay an additional recovery cost:

$$\frac{C(\delta)}{C_{\mathrm{rec}}} = r + \big(1 - \mathrm{Cov}(\delta)\big).$$

It reduces cost only if $\mathrm{Cov}(\delta) > r$; since $\mathrm{Cov}(\delta) \le m$, no risk-feasible policy can save cost when $r \ge m$.

4 EXPERIMENTS

4.1 EVALUATION PROTOCOL

Paired evaluation. At each controlled checkpoint, we run the memory and recovery routes from the same state, with the backbone, decoding protocol, and evaluator held fixed. The memory-route outcome defines R_{mem} under a predefined workload-specific criterion, while the paired recovery outcome supports repair and harm diagnostics. Because decisions within an episode, persona, source, or book can share history or substantially overlapping evidence, we average them within that unit and weight units equally; queries from the same source share a unit. Predictor-training, calibration, and final-test units are disjoint, and scorers are frozen before calibration (Appendix B).

Workloads and models. Primary fixed-state workloads are τ^3-Bench Retail (Yao et al., 2024; Barres & Shi, 2026), PersonaMem-v2 (Jiang et al., 2025), and PrefEval (Zhao et al., 2025); τ^3-Bench Airline and ALFWorld (Shridhar et al., 2021) provide low- and high-sufficiency regimes, while NarrativeXL (Moskvichev & Mai, 2023) tests long histories beyond the served context. Memory uses Mem0 (Chhikara et al., 2025) with a Llama-3.1-8B-Instruct (Grattafiori et al., 2024) writer. Downstream backbones are Llama-3.1-8B-Instruct, Mistral-7B-Instruct-v0.3 (Mistral AI, 2024), Gemma-3-12B-it (Gemma Team et al., 2025), and Qwen3.8-27B (Qwen Team, 2026); NarrativeXL excludes Mistral, which fails the output-format gate, and ALFWorld uses Qwen3-32B-AWQ (Yang et al., 2025; Qwen Team, 2025). Within each workload, all backbones share the same evidence, checkpoints, and splits, while branch outcomes and fitted scorers are backbone-specific; host-memory comparisons refit and recalibrate per host.

Table 1: False-safe risk and trust coverage across workloads and downstream backbones. Reference policies report FS / Cov.; risk-target policies report FS / Cov. / Exc. Arrows indicate the preferred direction. Bold marks the metric-wise best among the three risk-target policies at each backbone, workload, and α (ties included). Values are point estimates; 95% intervals are reported in Appendix B.3. †Zero-shot routing decisions are shared across backbones and scored against each backbone's own memory-route labels.

Llama-3.1-8B

Rule	α	Retail	PersonaMem-v2	PrefEval
Reference policies		FS↓ / Cov.↑	FS↓ / Cov.↑	FS↓ / Cov.↑
Always Trust		.618 / 1.000	.717 / 1.000	.550 / 1.000
Always Recover		.000 / .000	.000 / .000	.000 / .000
Fixed-F1		.531 / .902	.685 / .961	.399 / .788
TierMem router†		.052 / .064	.032 / .053	.015 / .020
Rule	α	FS↓ / Cov.↑ / Exc.↓	FS↓ / Cov.↑ / Exc.↓	FS↓ / Cov.↑ / Exc.↓
Empirical-risk	.05	**.024** / **.046** / .205	.062 / **.094** / .585	.060 / **.130** / .465
	.10	.101 / **.214** / .545	.104 / **.153** / .540	**.091** / **.203** / .530
	.20	.151 / **.321** / .110	.171 / **.247** / .470	.176 / **.384** / .480
Random + CRC	.05	.029 / .044 / **.000**	**.003** / .003 / **.000**	**.051** / .095 / **.375**
	.10	**.072** / .115 / **.000**	**.049** / .052 / **.040**	.108 / .185 / **.430**
	.20	**.133** / .207 / **.000**	.195 / .244 / **.150**	.210 / .378 / .465
ReCoverMem + CRC	.05	**.024** / **.046** / **.000**	.017 / .017 / .005	.053 / .122 / .380
	.10	.075 / .182 / .020	.066 / .101 / .125	**.091** / .201 / .445
	.20	.149 / .317 / .015	**.143** / .209 / .265	**.174** / .377 / **.440**

Mistral-7B

Rule	α	Retail	PersonaMem-v2	PrefEval
Reference policies		FS↓ / Cov.↑	FS↓ / Cov.↑	FS↓ / Cov.↑
Always Trust		.454 / 1.000	.682 / 1.000	.601 / 1.000
Always Recover		.000 / .000	.000 / .000	.000 / .000
Fixed-F1		.321 / .667	.669 / .983	.335 / .645
TierMem router†		.044 / .064	.032 / .053	.013 / .020
Rule	α	FS↓ / Cov.↑ / Exc.↓	FS↓ / Cov.↑ / Exc.↓	FS↓ / Cov.↑ / Exc.↓
Empirical-risk	.05	.043 / **.063** / .285	.056 / **.085** / .420	.051 / **.115** / .435
	.10	.085 / **.178** / .380	.116 / **.162** / .430	.097 / **.218** / .435
	.20	.199 / **.394** / .180	.239 / **.337** / .475	.196 / **.408** / .480
Random + CRC	.05	.022 / .060 / **.000**	.003 / .003 / **.000**	.060 / .102 / .365
	.10	**.058** / .142 / **.005**	**.042** / .059 / .065	.112 / .181 / .410
	.20	**.158** / .336 / **.005**	**.167** / .244 / **.100**	.223 / .377 / **.380**
ReCoverMem + CRC	.05	**.011** / .019 / **.000**	**.000** / .003 / **.000**	**.046** / .106 / **.345**
	.10	.064 / .114 / .015	.063 / .096 / **.050**	**.095** / .216 / **.350**
	.20	.166 / .341 / .025	.221 / .299 / .190	**.192** / .395 / .465

Gemma-3-12B

Rule	α	Retail	PersonaMem-v2	PrefEval
Reference policies		FS↓ / Cov.↑	FS↓ / Cov.↑	FS↓ / Cov.↑
Always Trust		.517 / 1.000	.645 / 1.000	.539 / 1.000
Always Recover		.000 / .000	.000 / .000	.000 / .000
Fixed-F1		.437 / .910	.645 / 1.000	.486 / .914
TierMem router†		.040 / .064	.032 / .053	.011 / .020
Rule	α	FS↓ / Cov.↑ / Exc.↓	FS↓ / Cov.↑ / Exc.↓	FS↓ / Cov.↑ / Exc.↓
Empirical-risk	.05	.048 / **.128** / .410	.084 / **.150** / .515	**.060** / **.117** / .500
	.10	.090 / **.229** / .330	.133 / **.230** / .505	**.102** / .197 / .505
	.20	.187 / **.452** / .280	.242 / **.367** / .510	**.163** / .318 / .495
Random + CRC	.05	.036 / .060 / **.000**	**.003** / .003 / **.000**	.066 / .115 / **.330**
	.10	**.068** / .134 / **.005**	**.045** / .059 / .075	.130 / **.229** / **.355**
	.20	**.168** / .328 / **.015**	**.175** / .289 / **.180**	.225 / **.417** / **.385**
ReCoverMem + CRC	.05	**.020** / .063 / **.000**	.021 / .028 / **.000**	**.060** / **.117** / .405
	.10	.075 / .188 / .015	.098 / .168 / **.040**	**.102** / .188 / .470
	.20	.177 / .427 / .040	.203 / .318 / .195	**.163** / .318 / .480

Qwen3.8-27B

Rule	α	Retail	PersonaMem-v2	PrefEval
Reference policies		FS↓ / Cov.↑	FS↓ / Cov.↑	FS↓ / Cov.↑
Always Trust		.542 / 1.000	.616 / 1.000	.636 / 1.000
Always Recover		.000 / .000	.000 / .000	.000 / .000
Fixed-F1		.501 / .915	.566 / .914	.627 / .985
TierMem router†		.054 / .064	.021 / .053	.016 / .020
Rule	α	FS↓ / Cov.↑ / Exc.↓	FS↓ / Cov.↑ / Exc.↓	FS↓ / Cov.↑ / Exc.↓
Empirical-risk	.05	**.023** / **.051** / .290	.066 / **.139** / .565	.051 / **.104** / .475
	.10	.077 / **.201** / .235	.126 / **.223** / .570	.117 / **.218** / .460
	.20	.212 / **.460** / .620	.227 / **.404** / .570	.194 / .346 / .465
Random + CRC	.05	.027 / .050 / **.000**	**.003** / .003 / **.000**	.055 / .102 / **.300**
	.10	**.057** / .121 / **.000**	**.028** / .052 / **.030**	**.115** / .190 / **.390**
	.20	**.129** / .235 / **.000**	**.161** / .265 / **.085**	.229 / **.380** / **.410**
ReCoverMem + CRC	.05	**.023** / **.051** / **.000**	.007 / .007 / .005	**.049** / .102 / .385
	.10	.061 / .161 / **.000**	.066 / .139 / .065	.117 / .216 / .430
	.20	.187 / .413 / .180	.186 / .338 / .180	**.188** / .338 / .430

Baselines and metrics. ALWAYS TRUST and ALWAYS RECOVER provide endpoint references; zero-shot TierMem (Zhu et al., 2026) and Fixed-F1 are uncalibrated baselines. To isolate the two components, Empirical-risk uses the learned score without the CRC correction, while Random + CRC keeps CRC but replaces the score with a frozen random ranking. Primary metrics are unit-equal FS and trust coverage; exceedance (Exc.) measures how often empirical FS exceeds the target under repeated calibration. Costs are normalized to a frozen recovery reference, using input tokens except on NarrativeXL, which counts executed-branch input and output tokens, so the two cost conventions are not directly comparable. Confidence intervals resample whole units with pairing preserved (Appendix D).

4.2 RISK CONTROL AND COVERAGE

Table 1 reports the primary fixed-state results across three workloads and four downstream backbones.**(1) ReCoverMem cuts false-safe risk 5–11-fold.** At $\alpha = .10$, ReCoverMem reduces empirical FS from .454–.717 under ALWAYS TRUST to .061–.117 across all 12 workload–backbone configurations, while still trusting 9.6–21.6% of decisions. Zero-shot TierMem trusts only 2.0–6.4% of decisions at a fixed operating point; ReCoverMem instead exposes α as an explicit, user-set risk budget. **(2) The CRC correction moves every configuration to a safer operating point.** At $\alpha = .10$, adding CRC to the same frozen learned score yields no higher empirical FS or coverage than Empirical-risk in any of the 12 configurations, and lower Exc. in all 12. On Retail/Llama, FS/Cov. moves from .101/.214 to .075/.182, while Exc. falls from .545 to .020. The correction trades a modest amount of coverage for this margin; consistent with the theory, the guarantee bounds expected risk rather than requiring every realized test set to meet the target.**(3) Learned ranking retains more coverage under the same correction.** Under identical CRC, ReCoverMem achieves higher point-estimate coverage than Random + CRC in 10 of 12 configurations. The paired gain is statistically resolved on Retail/Llama (+6.7 points), PersonaMem-v2/Gemma (+10.9), and PersonaMem-v2/Qwen (+8.7) (Table A8, Appendix B.3); the exceptions are Retail/Mistral (.114 versus .142) and PrefEval/Gemma (.188 versus .229). Although realized FS also differs across poli-

cies, this pattern supports the role of memory-sufficiency ranking in converting a fixed risk budget into usable trust. Additional scorer comparisons and TrustedError diagnostics appear in Appendices E.2 and D.

Sufficiency extremes follow the risk–coverage geometry. The bound in Section 3.3 ties attainable coverage to memory-sufficiency prevalence. At the high-sufficiency boundary, ALFWorld has $\widehat{\pi} = .963$: ALWAYS TRUST already attains empirical FS .037, and ReCoverMem matches that FS while retaining coverage .886, correctly recognizing that little recovery is needed. At the low-sufficiency boundary, Airline has $\widehat{\pi} = .128$ and a small calibration pool; at $\alpha = .10$ the finite-sample correction selects the always-recover fallback, a conservative choice that the geometric bound itself does not force. Figure 3 also shows an oracle-label diagnostic using held-out R_{mem} outcomes as scores to separate limits imposed by the risk and coverage geometry from limits of the learned ranking.

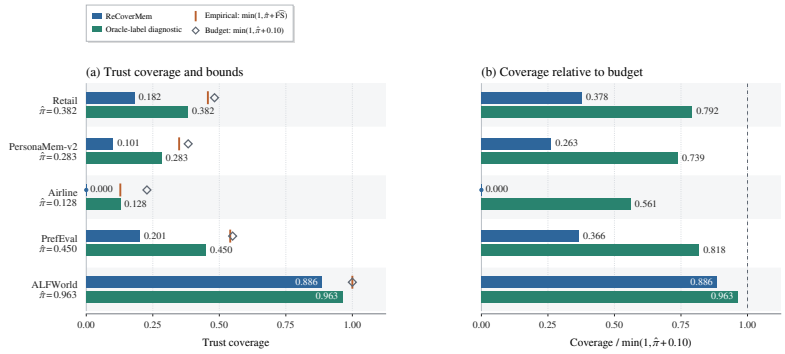


Figure 3: Risk–coverage operating regimes at $\alpha = .10$. Results use Llama-3.1-8B, except ALFWorld, which uses Qwen3-32B-AWQ. (a) compares ReCoverMem with a recalibrated oracle-label diagnostic, whose binary scores yield coverage $\widehat{\pi}$. Empirical $\min(1, \widehat{\pi} + \widehat{\text{FS}})$ and nominal $\min(1, \widehat{\pi} + .10)$ coverage references are also shown. (b) shows coverage normalized by the nominal reference. References use full-precision metrics and are descriptive rather than finite-test certificates.

4.3 EFFICIENCY AND DEPLOYMENT

Long-history routing reduces token cost. NarrativeXL evaluates the long-history regime absent from Retail: 99% of histories exceed the served 32K context, making recovery substantially more expensive. At $\alpha = .10$, ReCoverMem reduces executed-branch tokens by 8.5–13.6% relative to ALWAYS RECOVER, with all paired intervals above zero, while achieving empirical FS of .057–.085 and trust coverage of 11.7–18.2% (Table 2). This saving comes with a 3.0–7.7 point drop in routed success. At the stricter $\alpha = .05$, Gemma still saves 3.9% ($[2.5, 5.3]$) with no statistically detectable success change (-0.5 points, $[-1.5, 0.5]$) (Figure 4). Compared with Random + CRC, ReCoverMem achieves lower FS, higher coverage, and greater savings on all three backbones.

Table 2: Candidate-free routing on NarrativeXL at $\alpha = .10$. Brackets show 95% book-bootstrap intervals over 50 test books.

Backbone	Policy	FS	Cov.	Succ.	Saving (%)
Qwen3.8-27B	Random + CRC	.075 [.057,.093]	.108 [.087,.130]	**.740** [.703,.775]	7.9 [6.3,9.6]
	ReCoverMem + CRC	**.065** [.047,.087]	**.117** [.090,.145]	.737 [.698,.778]	**8.5** [6.5,10.5]
Llama-3.1-8B	Random + CRC	.090 [.070,.112]	.143 [.118,.170]	**.740** [.702,.775]	10.7 [8.8,12.7]
	ReCoverMem + CRC	**.085** [.065,.107]	**.150** [.123,.182]	.715 [.673,.753]	**11.1** [9.2,13.5]
Gemma-3-12B	Random + CRC	.070 [.053,.088]	.152 [.125,.180]	**.750** [.715,.783]	11.4 [9.4,13.6]
	ReCoverMem + CRC	**.057** [.038,.078]	**.182** [.152,.217]	**.750** [.703,.788]	**13.6** [11.3,16.3]

On-policy rollouts point to recovery as the next bottleneck. On Retail, ReCoverMem attains the lowest FS among threshold-based policies: .013 at coverage .104, versus .025/.129 for Empirical-risk and .113/.175 for Random + CRC. On Airline tasks, it recovers at every decision and precisely

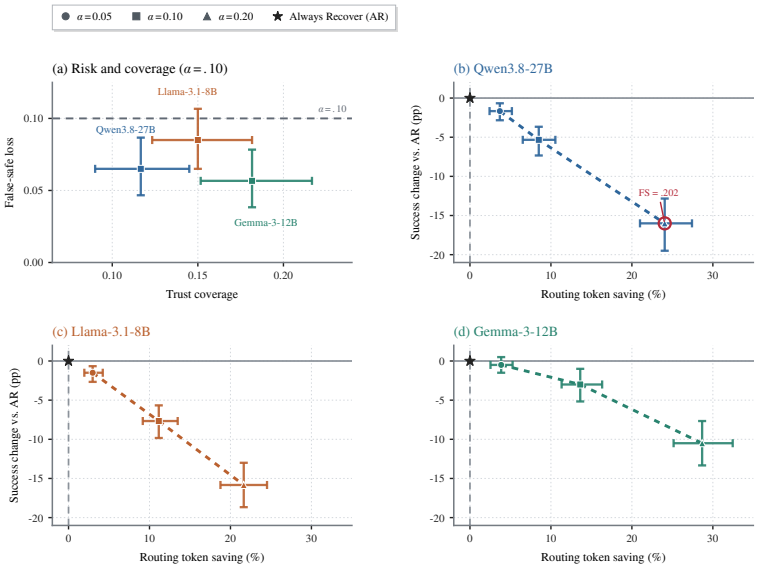


Figure 4: Candidate-free routing on NarrativeXL. **(a)** ReCoverMem FS and trust coverage at $\alpha = .10$; the dashed line marks the target. **(b–d)** Token savings and success change relative to ALWAYS RECOVER across risk targets. The red outline marks the only operating point above its target (Qwen, $\alpha = .20$, FS $= .202$). Error bars are 95% paired book-bootstrap intervals.

matches ALWAYS RECOVER.Final Retail task success remains zero for all policies, showing that better routing alone does not solve the end-to-end task. Fixed-state Retail/Llama diagnostics point to recovery as the next bottleneck: recovery repairs 6.6% of memory-route failures but harms 15.8% of memory-route successes. These small on-policy cohorts visit policy-dependent states, so we treat them as deployment diagnostics rather than part of the fixed-state guarantee (Appendix F.2).

4.4 ABLATION STUDIES

Flattening units into decisions. Flattening dependent decisions into independent calibration examples changes two things at once. With n units and m decisions, *Formal* uses unit-equal loss and the n-based CRC correction, *Diagnostic A* substitutes m for n, and *Naive* further switches to decision-equal loss. Formal–Diagnostic A therefore isolates the count and Diagnostic A–Naive the weighting; neither diagnostic is a certified procedure.

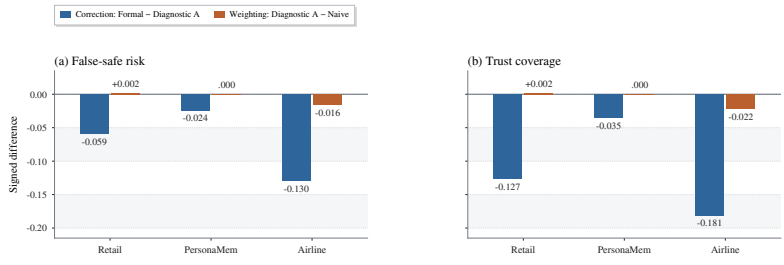


Figure 5: **Calibration-unit ablation.** Llama-3.1-8B-Instruct at $\alpha = .10$ and $n_{\text{cal}} = 20$.

Takeaway. The finite-sample correction is driven primarily by the number of independent calibration units, rather than by within-unit weighting (Figure 5).At $n = 20$, the correction term is $1/21 \approx .048$; treating decisions as independent would shrink it to .004–.007. On Retail, changing the count shifts FS/Cov. by $-.059/-.127$, compared with only $+.002/+.002$ from changing the weighting; Airline shows an even larger count effect, while PersonaMem-v2 shows no weighting difference at the reported precision. Duplicating decisions within a unit leaves the formal procedure unchanged: at $n_{\text{cal}} = 25$, test FS remains .060 from $1\times$ to $10\times$ replication, whereas the decision-count diagnostic rises to .106 at $2\times$. Conversely, with 400 independent PrefEval sources, CRC nearly matches Empirical-risk at $\alpha = .10$ (.091/.201 versus .091/.203 FS/Cov.), showing that the finite-sample cost becomes negligible when calibration units are plentiful (Appendix G.1).

Swapping the host memory. We vary the host memory on Retail/Llama with the split and downstream backbone fixed, refitting the scorer and recalibrating on each host's own memory-route labels. Table 3 reports the resulting operating points at $\alpha = .10$ under episode-equal weighting.

Takeaway. Risk control is preserved under host substitution; ranking advantage is not. Changing the host alters the labels, sufficiency prevalence, and score distribution, yet all four recalibrated policies attain empirical FS below target with exceedance at most .020. Paired coverage differences against Random + CRC exclude zero for Mem0 (+.067), Rolling Summary (+.041), and Three-Layer Memory (+.057), but not for A-Mem (−.035), where ReCoverMem instead attains lower FS (.034 vs. .063) at a stricter operating point. Prevalence does not determine that point: A-Mem and Mem0 differ by .003 in $\widehat{\pi}$ but .117 in coverage. Because each host is fitted independently and may induce a distinct decision distribution, the results demonstrate host-specific reinstantiation rather than transfer of a shared scorer across hosts.

Table 3: Host-memory sensitivity on Retail at $\alpha = .10$. Brackets give 95% intervals.

Host	Rule	$\widehat{\pi}$	FS	Cov.	Exc.
Mem0	Always Trust	.382	.618 [.508,.717]	1.000 [1.000,1.000]	–
	Random + CRC	.382	.072 [.050,.094]	.115 [.089,.144]	.000 [.000,.018]
	ReCoverMem + CRC	.382	.075 [.042,.113]	.182 [.139,.232]	.020 [.005,.050]
Rolling Summary	Always Trust	.387	.613 [.503,.711]	1.000 [1.000,1.000]	–
	Random + CRC	.387	.076 [.053,.099]	.126 [.101,.153]	.000 [.000,.018]
	ReCoverMem + CRC	.387	.058 [.035,.081]	.167 [.135,.207]	.000 [.000,.018]
A-Mem	Always Trust	.385	.615 [.502,.710]	1.000 [1.000,1.000]	–
	Random + CRC	.385	.063 [.040,.086]	.101 [.075,.124]	.000 [.000,.018]
	ReCoverMem + CRC	.385	.034 [.013,.060]	.065 [.037,.098]	.000 [.000,.018]
Three-Layer Memory	Always Trust	.346	.654 [.542,.756]	1.000 [1.000,1.000]	–
	Random + CRC	.346	.078 [.054,.101]	.111 [.084,.140]	.000 [.000,.018]
	ReCoverMem + CRC	.346	.049 [.029,.069]	.168 [.140,.205]	.000 [.000,.018]

Removing candidate generation. The candidate-based scorer requires a memory-route candidate to be generated before the routing decision. The candidate is reused when the controller selects TRUST, but its generation cost is incurred unnecessarily when the controller selects RECOVER. At the Retail/Llama operating point with coverage .182, this occurs for 81.8% of decisions. Relative to an ALWAYS RECOVER reference normalized to one, candidate-based routing incurs input costs of 1.680 on PersonaMem-v2 and 1.549 on PrefEval.

Table 4: Candidate-generation ablation on fixed-state Retail.

Condition	FS	Cov.	Routed succ.	Norm. input cost
A. Candidate-based	.075 [.042,.113]	.182 [.139,.232]	.356 [.250,.475]	1.781 [1.732,1.822]
B. Same routes	.075 [.042,.113]	.182 [.139,.232]	.356 [.250,.475]	.994 [.991,.996]
C. Candidate-free	.060 [.031,.093]	.152 [.122,.192]	.356 [.250,.475]	.995 [.994,.997]

Takeaway. Table 4 isolates the candidate-generation overhead on fixed-state Retail/Llama. Condition B preserves A's routing decisions while excluding the cost of candidates discarded on RECOVER; risk, coverage, and routed success are therefore unchanged by construction. Normalized input cost decreases from 1.781 to .994 ($\Delta = .787$, 95% paired CI $[.740, .827]$), showing that discarded candidate generation accounts for essentially all of the excess cost. The deployable candidate-free controller C removes this overhead at a comparable operating point, achieving FS/Cov. .060/.152 at cost .995. Further savings on Retail are inherently limited: ALWAYS TRUST costs .949 relative to the recovery reference, so even perfect ranking can save at most $(\widehat{\pi} + \alpha)(1 - .949) \approx 2.4\%$. Retail therefore characterizes a regime with limited cost headroom, whereas NarrativeXL (Section 4.3) shows the benefit of selective routing when branch costs differ substantially.

5 CONCLUSION

We introduced ReCoverMem, a TRUST/RECOVER controller that combines memory-sufficiency scoring with unit-level conformal risk control. Under exchangeability, it controls expected unit-averaged false-safe loss while using learned ranking to retain trust coverage. Across workloads and backbones, our results show that calibration-unit count, ranking quality, recovery effectiveness, and branch cost jointly determine the practical value of selective memory trust. Further limitations are discussed in Appendix H.

Reproducibility Statement

We provide detailed specifications for reproducing the ReCoverMem experiments and statistical analyses. Section 4.1 describes the workloads, models, statistical units, baselines, routing protocols, metrics, and cost accounting used in the main evaluation. Appendix B documents unit construction, eligibility audits, and predictor-training, calibration, and final-test splits. Appendix C specifies the memory-sufficiency scorers, feature definitions, preprocessing and fitting procedures, model and baseline configurations, and frozen implementation details. Appendix D gives the exact definitions of unit-equal metrics, cost measures, and uncertainty procedures, while Appendix B.3 reports uncertainty intervals for the main results. Predictor-training, calibration, and final-test units are kept disjoint, and fitted scorers are frozen before calibration. Where applicable, we report model revisions, benchmark revisions, serving configurations, random seeds, evidence budgets, and resampling procedures needed to reconstruct the reported experiments.

Ethics Statement

This work studies reliability in memory-augmented LLM agents, with the goal of reducing decisions made from insufficient compressed memory. Risk control over false-safe decisions should not be interpreted as a general safety guarantee: the stated conformal guarantee depends on exchangeability of calibration and future statistical units, controls the specified false-safe loss rather than all downstream errors, and does not automatically extend to distribution shifts or routing-induced on-policy trajectories. Recovery itself can also fail or degrade an otherwise successful memory-route decision. We therefore report false-safe risk, trust coverage, recovery behavior, downstream success, and execution cost separately, and explicitly characterize settings in which the assumptions or operating regime make selective trust uninformative. The experiments use existing research benchmarks and are intended to support reliability evaluation and safer deployment of memory-augmented agents.

AI Use Statement

Generative AI tools were used during this work for language editing and clarification, assistance with drafting and debugging experimental and analysis code, and discussion of experimental design and presentation. They were not treated as sources of experimental evidence or as substitutes for the reported measurements. The authors reviewed AI-assisted text, inspected and tested AI-assisted code, and verified reported numerical results against the corresponding experimental logs, evaluation artifacts, and analysis outputs. Literature claims and citations were checked against the underlying source materials rather than accepted from generated text without verification. The authors take responsibility for the final content of this work, including text, claims, code, and other artifacts produced with the aid of generative AI.

References

Yasin Abbasi Yadkori, Ilja Kuzborskij, David Stutz, András György, Adam Fisch, Arnaud Doucet, Iuliya Beloshapka, Wei-Hung Weng, Yao-Yuan Yang, Csaba Szepesvári, Ali Taylan Cemgil, and Nenad Tomasev. Mitigating LLM hallucinations via conformal abstention. *arXiv preprint arXiv:2405.01563*, 2024.

Mayur Akewar and Ravi Ranjan. SafeCommit: Certifying when memory-grounded agents may safely act. *arXiv preprint arXiv:2608.04289*, 2026.

Anastasios N. Angelopoulos and Stephen Bates. Conformal prediction: A gentle introduction. *Foundations and Trends in Machine Learning*, 16(4):494–591, 2023. doi: 10.1561/2200000101.

Anastasios N. Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal risk control. In *International Conference on Learning Representations*, 2024.

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

Victor Barres and Ben Shi. τ^3-Bench: Fixing airline + retail. Official τ-bench blog, February 2026. URL https://taubench.com/blog/tau3-task-fixes.html.

Lingjiao Chen, Matei Zaharia, and James Zou. FrugalGPT: How to use large language models while reducing cost and improving performance. *Transactions on Machine Learning Research*, 2024.

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

C. K. Chow. On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, 16(1):41–46, 1970. doi: 10.1109/TIT.1970.1054406.

Xinle Deng, Ruobin Zhong, Hujin Peng, Xiaoben Lu, Yanzhe Wu, Guang Li, Buqiang Xu, Yunzhi Yao, Jizhan Fang, Haoliang Cao, Junjie Guo, Yuan Yuan, Ziqing Ma, Yuanqiang Yu, Rui Hu, Baohua Dong, Hangcheng Zhu, and Ningyu Zhang. MemTrace: Tracing and attributing errors in large language model memory systems. *arXiv preprint arXiv:2605.28732*, 2026.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems*, 2017.

Yonatan Geifman and Ran El-Yaniv. SelectiveNet: A deep neural network with an integrated reject option. In *International Conference on Machine Learning*, 2019.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025. URL https://arxiv.org/abs/2503.19786.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. URL https://arxiv.org/abs/2407.21783.

Xixuan Hao, Zeyu Zhang, Zehao Lin, Yihang Sun, Ziliang Guo, Xichong Zhang, Yuxuan Liang, Feiyu Xiong, and Zhiyu Li. MemOps: Benchmarking lifecycle memory operations in long-horizon conversations. *arXiv preprint arXiv:2607.12893*, 2026.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong C. Park. Adaptive-RAG: Learning to adapt retrieval-augmented large language models through question complexity. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics*, 2024.

Bowen Jiang, Yuan Yuan, Maohao Shen, Zhuoqun Hao, Zhangchen Xu, Zichen Chen, Ziyi Liu, Anvesh Rao Vijjini, Jiashu He, Hanchao Yu, Radha Poovendran, Gregory Wornell, Lyle Ungar, Dan Roth, Sihao Chen, and Camillo Jose Taylor. PersonaMem-v2: Towards personalized intelligence via learning implicit user personas and agentic memory. *arXiv preprint arXiv:2512.06688*, 2025. URL https://arxiv.org/abs/2512.06688.

Dongming Jiang, Yi Li, and Bingzhe Li. Jev-Mem: System-one-controlled agentic memory for efficient AI agents. *arXiv preprint arXiv:2609.23986*, 2026.

Zhengbao Jiang, Frank F. Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. Active retrieval augmented generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

Hailey Joren, Jianyi Zhang, Chun-Sung Ferng, Da-Cheng Juan, Ankur Taly, and Cyrus Rashtchian. Sufficient context: A new lens on retrieval augmented generation systems. In *International Conference on Learning Representations*, 2025. arXiv:2411.06037.

Saurav Kadavath et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *International Conference on Learning Representations*, 2023.

Kuang-Huei Lee, Xinyun Chen, Hiroki Furuta, John Canny, and Ian Fischer. A human-inspired reading agent with gist memory of very long contexts. In *International Conference on Machine Learning*, 2024.

Yonghoon Lee, Rina Foygel Barber, and Rebecca Willett. Distribution-free inference with hierarchical data. *arXiv preprint arXiv:2306.06342*, 2023.

Xianxuan Long, Zhikai Chen, Shenglai Zeng, Shouren Wang, Kai Guo, and Jiliang Tang. MemTrace: Probing what final accuracy misses in long-term memory. *arXiv preprint arXiv:2606.17328*, 2026.

Enze Ma, Yufan Zhou, Wei-Chieh Huang, Jie Yang, Huanhuan Ma, Zixuan Wang, Chengze Li, Chunyu Miao, Philip S. Yu, and Zhen Wang. MemAudit: Auditing long-term agent memory via hidden user-state recovery. *arXiv preprint arXiv:2606.24595*, 2026.

David Madras, Toniann Pitassi, and Richard S. Zemel. Predict responsibly: Improving fairness and accuracy by learning to defer. In *Advances in Neural Information Processing Systems*, 2018.

Mistral AI. Mistral-7B-Instruct-v0.3. Hugging Face model card, 2024. URL https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3. Accessed September 25, 2026.

Christopher Mohri and Tatsunori Hashimoto. Language models with conformal factuality guarantees. In *International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 36029–36047, 2024. arXiv:2402.10978.

Arsenii Moskvichev and Ky-Vinh Mai. NarrativeXL: A large-scale dataset for long-term memory models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 15058–15072, 2023. arXiv:2305.13877.

Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E. Gonzalez, M. Waleed Kadous, and Ion Stoica. RouteLLM: Learning to route LLMs with preference data. In *International Conference on Learning Representations*, 2025.

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

Victor Quach, Adam Fisch, Tal Schuster, Adam Yala, Jae Ho Sohn, Tommi S. Jaakkola, and Regina Barzilay. Conformal language modeling. In *International Conference on Learning Representations*, 2024. arXiv:2306.10193.

Qwen Team. Qwen3-32B-AWQ. Hugging Face model card, 2025. URL https://huggingface.co/Qwen/Qwen3-32B-AWQ. Accessed September 25, 2026.

Qwen Team. Qwen3.8-27B. Hugging Face model card, 2026. URL https://huggingface.co/Qwen/Qwen3.8-27B. Accessed September 25, 2026.

Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew Hausknecht. ALFWorld: Aligning text and embodied environments for interactive learning. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=0IOX0YcCdTn.

Wenwen Si, Sooyong Jang, Insup Lee, and Osbert Bastani. Conformal constrained policy optimization for cost-effective LLM agents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 25446–25453, 2026.

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. Springer, 2005.

An Yang et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. τ-bench: A benchmark for tool-agent-user interaction in real-world domains. *arXiv preprint arXiv:2406.12045*, 2024.

Siyan Zhao et al. Do LLMs recognize your preferences? evaluating personalized preference following in LLMs. In *International Conference on Learning Representations*, 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.

Qiming Zhu, Shunian Chen, Rui Yu, Zhehao Wu, and Benyou Wang. From lossy to verified: A provenance-aware tiered memory for agents. *arXiv preprint arXiv:2602.17913*, 2026.

APPENDIX

A CALIBRATION AND RISK–COVERAGE DETAILS

A.1 UNIT-LEVEL CRC AND BOUNDARY CASES

Fixed threshold family. For a given scorer and evaluation protocol, each unit has a finite set of controlled decisions with fixed scores and memory-route outcomes. Changing τ changes only whether each decision is trusted. Consequently,

$$L_i(\tau) = \frac{1}{|\mathcal{T}_i|} \sum_{t \in \mathcal{T}_i} \mathbb{1}[s_{i,t} > \tau](1 - R_{\mathrm{mem},i,t})$$

is bounded in $[0, 1]$ and non-increasing. Under the strict rule, a decision ceases to be trusted at $\tau = s_{i,t}$, making the loss right-continuous. The displayed average applies to non-empty units. Retained empty units have identically zero loss. Scores are finite, so the thresholds $-\infty$ and $+\infty$ implement Always Trust and Always Recover, respectively.

The empirical loss changes only at observed calibration scores. Searching their distinct values together with $-\infty$ and $+\infty$ therefore implements the infimum-based CRC rule. Scores tied at the selected threshold are all routed to RECOVER; no arbitrary ordering within a tie is required.

Finite-sample correction and fallback. For the loss bound $B = 1$, the corrected empirical criterion is

$$\frac{n}{n+1} \widehat{L}_n(\tau) + \frac{1}{n+1} \leq \alpha. \tag{2}$$

Its feasible set is empty exactly when

$$\frac{1}{n+1} > \alpha.$$

Necessity follows because empirical loss is nonnegative. Conversely, if $1/(n+1) \leq \alpha$, then $+\infty$ is feasible because it has zero empirical loss.

When the feasible set is empty, the procedure selects $+\infty$. This policy never trusts and has identically zero false-safe loss. Its zero loss does not assert that recovery succeeds or that the resulting trajectory is free of other errors. A finite threshold can also yield zero coverage on a particular test set without implying zero coverage on every future unit.

Scope of the guarantee. Under the exchangeability assumption stated in Section 3.2, the bounded, monotone, right-continuous unit losses satisfy the CRC conditions (Angelopoulos et al., 2024).

The expectation averages over calibration data and a future unit, conditional on the frozen scorer and decision-generation protocol. It is not a guarantee conditional on every realized calibration sample. It also does not bound the probability that empirical test risk exceeds α. The Exc. diagnostic measures such empirical exceedance under the specified resampling protocol and is not a theorem-violation probability.

Empty units. The convention for excluding or retaining empty units is fixed before calibration and applied consistently to evaluation. If empty units are excluded, the target population is restricted to non-empty units and $\rho = 1$. If they are retained, their loss, trust coverage, and memory-sufficient fraction are all defined as zero. The same convention is used when computing every term in the risk–coverage relations.

Algorithm 1 ReCoverMem: Unit-Level Calibration and Routing

Require: Frozen scorer s or s_{cf}; variant $v \in \{\text{candidate}, \text{candidate-free}\}$; risk target $\alpha \in (0, 1)$
Require: Calibration records $\mathcal{D}_{\mathrm{cal}} = \{(s_{i,t}, R_{\mathrm{mem},i,t})_{t \in \mathcal{T}_i}\}_{i=1}^{n}$, $n \geq 1$, with finite scores

Offline calibration

procedure CALIBRATE($\mathcal{D}_{\mathrm{cal}}, \alpha$)
  $\mathcal{C} \leftarrow \{-\infty, +\infty\} \cup \{s_{i,t} : i = 1, \ldots, n,\ t \in \mathcal{T}_i\}$
  for each $\tau \in \mathcal{C}$ in increasing order **do**
    for $i = 1, \ldots, n$ **do**
      if $|\mathcal{T}_i| = 0$ **then**
        $L_i(\tau) \leftarrow 0$ ▷ Retained empty unit
      else
        $L_i(\tau) \leftarrow \dfrac{1}{|\mathcal{T}_i|} \displaystyle\sum_{t \in \mathcal{T}_i} \mathbb{1}[s_{i,t} > \tau \wedge R_{\mathrm{mem},i,t} = 0]$
      end if
    end for
    if $\dfrac{\sum_{i=1}^{n} L_i(\tau) + 1}{n+1} \leq \alpha$ **then**
      return τ ▷ Smallest feasible threshold
    end if
  end for
  return $+\infty$ ▷ Always-Recover fallback
end procedure
$\hat{\tau} \leftarrow$ CALIBRATE($\mathcal{D}_{\mathrm{cal}}, \alpha$)

Decision-time routing

procedure ROUTE($x_t, \mathcal{M}_t, \mathcal{H}_t, v, \hat{\tau}$)
  $E_t \leftarrow \mathcal{E}(x_t, \mathcal{M}_t)$
  if $v = \text{candidate}$ **then**
    $\tilde{a}_t \leftarrow$ MEMORYANSWER(x_t, E_t) ▷ Generate without executing
    $s_t \leftarrow s(x_t, E_t, \tilde{a}_t)$
  else
    $s_t \leftarrow s_{\mathrm{cf}}(x_t, E_t)$ ▷ No candidate generation
  end if
  if $s_t > \hat{\tau}$ **then**
    if $v = \text{candidate-free}$ **then**
      $\tilde{a}_t \leftarrow$ MEMORYANSWER(x_t, E_t)
    end if
    return (TRUST, $\tilde{a}_t$)
  else
    $a_t^{\mathrm{rec}} \leftarrow$ RECOVERYANSWER($x_t, \mathcal{H}_t$)
    return (RECOVER, a_t^{rec}) ▷ Ties recover
  end if
end procedure

Calibration and routing procedure. Algorithm 1 summarizes offline calibration and decision-time routing. The scorer, preprocessing, routing variant, and branch-generation protocol are fixed before calibration. Calibration records contain scores and memory-route outcomes grouped by statistical unit; the scores are computed using the same variant used at deployment. Memory-route outcomes are used only during calibration and evaluation, not as routing inputs.

Here, MEMORYANSWER generates the memory-route answer or action using the frozen downstream protocol. RECOVERYANSWER obtains historical evidence through the frozen recovery operator and generates the recovery-route answer or action. The host executes or returns only the selected output. Candidate-based routing reuses its existing candidate when trusting; candidate-free routing generates an answer only for the selected branch. The calibration count n follows the pre-specified unit-eligibility rule; excluded empty units are not included in $\mathcal{D}_{\mathrm{cal}}$.

The routing procedure can be applied at successive decision states, but the CRC guarantee requires exchangeability of the corresponding unit-level loss functions. The pseudocode does not extend that guarantee to arbitrary routing-induced state distributions.

A.2 Risk–Coverage Bound

For a non-empty unit i, recall its memory-sufficient fraction $p_i = |\mathcal{T}_i|^{-1} \sum_{t \in \mathcal{T}_i} R_{\mathrm{mem},i,t}$. Let $\rho_i = \mathbb{1}[|\mathcal{T}_i| > 0]$; retained empty units have $p_i = L_i = c_i = 0$. Splitting trusted decisions by memory-route outcome gives

$$c_i(\delta_\tau) = \frac{1}{|\mathcal{T}_i|} \sum_{t \in \mathcal{T}_i} \mathbb{1}[s_{i,t} > \tau] R_{\mathrm{mem},i,t} + L_i(\tau) \tag{3}$$

$$\leq p_i + L_i(\tau), \qquad c_i(\delta_\tau) \leq \rho_i. \tag{4}$$

The first line is evaluated on non-empty units; the inequalities also hold under the zero convention for empty units. Averaging under the same unit distribution yields

$$\mathrm{Cov}(\tau) \leq \min\{\rho, \pi + \mathrm{FS}(\tau)\}.$$

For a calibrated random threshold, averaging over calibration and a future unit similarly gives $\mathbb{E}[c_{\mathrm{test}}(\delta_{\hat{\tau}})] \leq \min\{\rho, \pi + \alpha\}$ under the CRC assumptions. This is not a conditional guarantee for every realized calibration sample.

For a fixed empirical population with consistent unit weighting,

$$\widehat{\mathrm{Cov}} \leq \min\{\widehat{\rho}, \widehat{\pi} + \widehat{\mathrm{FS}}\}. \tag{5}$$

Replacing $\widehat{\mathrm{FS}}$ by α in this empirical inequality additionally requires $\widehat{\mathrm{FS}} \leq \alpha$. The reference $\min(1, \widehat{\pi} + \alpha)$ used for non-empty populations is therefore neither a finite-test certificate nor a claim that a particular scorer can attain that coverage.

A.3 Random Scoring as a Calibration Control

CRC does not require probability-calibrated scores. A label-independent score fixed before calibration also defines a bounded, monotone loss family. Under the same unit-level exchangeability assumption, CRC can therefore calibrate it. Random + CRC separates the role of calibration from that of learned ranking: an informative ranking can retain more coverage at a given nominal target, but need not do so on every split.

B Calibration Units and Workload Audits

Grouping and statistical units. Calibration follows the sampling structure of each workload. Controlled decisions are grouped by episode in τ^3-Bench and ALFWorld, by persona in PersonaMem-v2, by source in PrefEval, and by book in NarrativeXL. Queries that reuse the same or substantially overlapping source material belong to the same source unit. Losses are averaged within units before units receive equal weight; individual decisions within a unit are not counted as additional calibration units.

Finite-sample feasibility. For n calibration units and losses bounded in $[0, 1]$, the CRC criterion is

$$\frac{n}{n+1} \widehat{L}_n(\tau) + \frac{1}{n+1} \leq \alpha. \tag{6}$$

Since $\widehat{L}_n(\tau) \geq 0$, feasibility requires

$$\frac{1}{n+1} \leq \alpha. \tag{7}$$

Thus, at least 19 calibration units are required for $\alpha = .05$ and 9 for $\alpha = .10$. These requirements concern the corrected calibration criterion. The ALWAYS RECOVER fallback incurs zero false-safe loss regardless of calibration sample size.

Evaluation splits. Table A1 reports statistical-unit counts. Predictor-training, calibration, and final-test units are disjoint. Retail has 40 designated calibration episodes, but two yield no controlled decisions; its non-empty CRC calibration pool therefore contains 38 episodes. This eligibility rule is fixed before threshold selection. Airline reuses the Retail-trained scorer and calibrates it on Airline episodes. NarrativeXL uses 12 questions per book, yielding 600 questions in each of its three 50-book splits.

Table A1: **Statistical units and evaluation splits.** P/C/T denotes predictor-training, calibration, and final-test units. Counts outside parentheses are designated split sizes; parenthetical counts denote non-empty units that contribute controlled decisions to the corresponding stage.

Workload	Unit	P	C	T
τ^3-Bench Retail	Episode	34	40 (38)	40
PersonaMem-v2	Persona	16	20	24
PrefEval	Source	24	400	547
τ^3-Bench Airline	Episode	N/A	25	18
ALFWorld	Episode	16 (7)	24 (18)	24 (18)
NarrativeXL	Book	50	50	50

Source eligibility audit. Nominal query counts can substantially exceed source-unit counts. Under our grouping rules, the evaluated MAB-AR configuration contains 700 queries but only 7 source units, while LongMemEval-V2 contains 451 queries but only 2 source units (Table A2). Neither configuration supplies enough units for our disjoint predictor-training, calibration, and final-test protocol at the primary risk targets. This exclusion concerns our protocol, not the suitability of these benchmarks for other evaluations.

PrefEval contains 985 audited source units. Its formal evaluation pool comprises 971 sources: 24 for predictor training, 400 for calibration, and 547 for final testing. The other 14 audited sources are outside this formal evaluation pool.

Table A2: **Source-unit eligibility under audited configurations.** Eligibility refers to the primary split and calibration protocol. Audited source counts precede selection of the formal evaluation pool.

Benchmark	Nominal queries	Source units	Eligibility
MAB-AR	700	7	Excluded
LongMemEval-V2	451	2	Excluded
PrefEval	1000	985	Included

PrefEval exceedance protocol. The primary threshold uses all 400 calibration sources. The Exc. diagnostic uses 200 calibration-pool resamples with 200 sources per replicate; each threshold is evaluated on the frozen 547-source test population. The diagnostic therefore uses a different calibration size from the primary operating point.

B.1 Source-Grouping Details

MemoryAgentBench Accurate Retrieval. The Accurate Retrieval portion of MemoryAgentBench (MAB-AR) contains 700 scored queries but only seven defensible source histories:

Source	Source histories	Queries
EventQA (`eventqa_full`)	5	500
RULER QA1 (`ruler_qa1_197K`)	1	100
RULER QA2 (`ruler_qa2_421K`)	1	100
Total	7	700

Each source history is reused for many queries against the same underlying context and compression outcome. Additional nominal EventQA samples were verified to be truncated prefixes of the same underlying books rather than new source histories.

With only seven source-level units, any disjoint train/calibration/test construction leaves fewer than the 19 calibration units required at $\alpha = 0.05$ and fewer than the 9 required at $\alpha = 0.10$. We therefore classify MAB-AR as ineligible for the primary CRC evaluation.

LongMemEval-V2. LongMemEval-V2 contains 451 nominal questions, but many questions reuse strongly overlapping source histories or trajectory pools. After accounting for this overlap, the only defensible non-overlapping source-level partition in our audit yields two units.

For $n = 2$, the finite-sample correction is

$$\frac{1}{n+1} = \frac{1}{3},$$

which exceeds all target risks considered in the paper. LongMemEval-V2 is therefore not used to support our CRC validity claim.

We additionally encountered judge-dependent and multimodal evaluation requirements in the released stack. These are secondary practical limitations; the exchangeable-unit count alone is sufficient for the exclusion.

PrefEval. The audited 1,000 instances resolve to 997 distinct preference–query pairings and 985 source components after merging reused preferences, queries, or pairings. Alternative formulations of the same underlying pairing do not create additional source units. Grouping avoids treating reused sources as additional units.

B.2 Workload Screening

The screening results below concern the evaluated agent, serving, and instrumentation configurations. They are not claims about every implementation or intended use of the underlying benchmarks.

Table A3: **Workload suitability and screening audit.** Disposition refers only to suitability for the present TRUST/RECOVER protocol, not to the general quality or usefulness of the benchmark. Diagnostic workloads are retained only for the specific property indicated below.

Environment	Disposition	Audit outcome
MAB-AR	Excluded	700 nominal queries correspond to only seven audited source histories, insufficient for the disjoint unit-level calibration protocol; see Appendix B.
LongMemEval-V2	Excluded	451 nominal queries correspond to only two audited source units; see Appendix B.
LoCoMo	Excluded	1,986 nominal queries correspond to only ten conversation histories. Any disjoint predictor-training, calibration, and final-test partition leaves fewer than the nine calibration units required at $\alpha = .10$. see Appendix B.
MemoryArena Bundled Shopping	Diagnostic only	Memory-route success is near the floor under both evaluated backbones ($\hat{\pi} = .0267$ with Llama and $.0133$ with Qwen). Under the Llama configuration, the recovery route succeeds on $0/75$ controlled decisions.
MemoryArena Progressive Search	Excluded	History is not load-bearing in the audited configuration: the no-history counterfactual succeeds on 80% of episodes versus 60% with history, despite substantial context pressure.
SWE-bench Verified	Diagnostic only	None of the 19 audited instances exceeds the effective context window; median history occupancy is approximately 46%. The configuration therefore does not provide natural compression pressure for the claim being tested.
AppWorld	Excluded	The evaluated open-weight base agent succeeds on $0/90$ tasks, so memory-route labels collapse near the failure floor independently of memory quality.
τ^3-Bench Telecom	Excluded	The pre-specified viability gate fails: only one of 171 controlled decisions is memory-route positive and the recovery route has zero successes. The domain additionally exposes a recoverability mismatch discussed below.
τ-Bench (original)	Excluded	Histories remain far below the available context window, while the user simulator is not reproducible under the evaluated configuration, preventing stable paired branch entry states.
InfiniteBench	Excluded	The audited configurations have low memory-sufficiency prevalence ($\hat{\pi} \approx .05$–$.13$), leaving little useful selective-trust coverage at the target risk levels.
Gaia2	Excluded	Most audited scenarios are effectively single-turn for the relevant evaluation, and the production path does not expose the deterministic programmatic per-decision signal required by our protocol.
WorkArena++	Excluded	The evaluated open-weight configuration reaches no controlled decisions from which paired memory and recovery outcomes can be constructed.
ToolSandbox / CRMArena-Pro	Excluded	The evaluated environments do not provide the seeded checkpoint-and-restore behavior required to construct identical paired branch entry states.
WebArena / OSWorld / AndroidWorld / AgentDojo / Mind2Web / BFCL	Excluded	The available evaluation signal is primarily terminal, yielding no repeated per-decision outcome structure for the unit-level routing analysis used here.

Context pressure does not imply memory dependence. MemoryArena Progressive Search provides a useful counterexample. Although the audited episodes exhibit substantial long-history pressure, removing accumulated history does not reduce task success: the no-history arm succeeds on

80% of the audited episodes versus 60% for the history-conditioned arm. Among the 12 episodes solved with history, 11 are also solved without it. Thus, a long context by itself does not establish that memory is load-bearing for the final decision.

The opposite issue occurs in SWE-bench Verified. Across the 19 audited instances, no history exceeds the effective context window, with median occupancy approximately 46%. The workload can still be useful for other agent evaluations, but this configuration does not isolate a natural need for history compression.

Base-agent competence and label prevalence matter. A selective memory controller is difficult to evaluate when the base agent succeeds almost nowhere, because $R_{\text{mem}} \approx 0$ independently of memory quality. This occurs in AppWorld, where the evaluated agent succeeds on $0/90$ tasks, and in MemoryArena Bundled Shopping. For the latter, changing from Llama to Qwen removes several infrastructure limitations but does not improve the relevant label distribution: observed memory-sufficiency prevalence decreases from .0267 to .0133. These configurations are therefore retained only as diagnostic evidence about the operating regime rather than as primary evaluations of ReCoverMem.

Low prevalence can also make an otherwise valid workload uninformative for selective trust. InfiniteBench exhibits $\hat{\pi}$ in the approximate range .05–.13 in the audited configurations. Under the risk–coverage relationship in Section 3.3, such populations leave little room for substantial trust coverage at small risk targets.

Telecom viability and recovery audit. The τ^3-Bench Telecom pilot contains 16 eligible episodes and 171 controlled decisions. It fails the pre-specified viability gate (Table A4), so the formal splits, generated before outcome inspection, were not activated. The joint memory/recovery outcome counts are $170/0/1/0$ for $00/01/10/11$. Thus, decision-pooled memory-route success is $1/171 \approx .0058$, and recovery succeeds on $0/171$ decisions. This pooled rate is not an episode-equal prevalence estimate.

Table A4: **Pre-specified Telecom pilot viability gate.** Counts refer to 171 controlled decisions in 16 eligible episodes.

Requirement	Minimum	Observed	Outcome
Memory-route positives	4	1	Fail
Memory-route negatives	10	170	Pass
Episodes containing a positive	3	1	Fail
Both outcome classes present	2	2	Pass

All 171 pairs pass the recorded checks for identical branch-entry states, evidence budgets, and provenance. These checks support paired comparability but do not establish that the available evidence is sufficient for successful recovery. All 82 audited gold actions are terminal remediation writes, and the observation licensing the required remediation is absent from the recorded trajectory in 11 of 16 pilot episodes. This suggests an information-acquisition limitation of the logging policy. The absence of observed recovery successes does not establish that recovery is impossible by construction, nor does an action's read/write category alone determine recoverability.

A subsequent audit corrected the host-budget measurement to use the 12-step logging trajectories rather than the shorter gold trajectories. Maximum packed native evidence was 938 tokens, giving the same 1,024-token evidence budget used in Retail; no pilot memory branch required truncation. The serialized `recovery_query` field was empty in both Telecom and Retail logs, although the actual state query was passed to recovery. This logging omission limits reconstruction from that field alone.

The pilot motivates reporting whether required evidence was observed before each scored decision. This is a post-pilot diagnostic recommendation, not an additional pre-registered eligibility criterion. The existing Airline audit also leaves domain-specific identifier coverage in its provenance matcher to be verified; it is not evidence of a confirmed Airline matching failure.

B.3 Uncertainty Quantification for Main Results

FS and coverage intervals below quantify final-test statistical-unit sampling uncertainty with the scorer and calibrated threshold held fixed. Exc. intervals quantify finite Monte Carlo error in the archived repeated-calibration diagnostic. All intervals are pointwise; they are not simultaneous across workloads, backbones, policies, risk targets, or metrics.

Table A5: **Retail uncertainty intervals.** FS/Cov. are pointwise 95% percentile bootstrap intervals over the 40 final-test episodes ($B = 1000$, seed 13), with the scorer and calibrated threshold held fixed. Exc. entries are pointwise exact 95% Clopper–Pearson Monte Carlo intervals over the archived $B = 200$ repeated-calibration replicates. Point estimates are reported in Table 1; only intervals are repeated here.

Llama-3.1-8B

Rule	α	FS 95% CI	Cov. 95% CI	Exc. 95% MC CI
Always Trust	–	[.514,.723]	[1.000,1.000]	–
Always Recover	–	[.000,.000]	[.000,.000]	–
Fixed-F1	–	[.433,.631]	[.833,.960]	–
TierMem router†	–	[.015,.104]	[.021,.120]	–
Empirical-risk	.05	[.007,.042]	[.022,.072]	[.151,.268]
Empirical-risk	.10	[.054,.153]	[.161,.268]	[.473,.615]
Empirical-risk	.20	[.089,.220]	[.253,.392]	[.070,.162]
Random + CRC	.05	[.013,.050]	[.024,.065]	[.000,.018]
Random + CRC	.10	[.049,.099]	[.087,.142]	[.000,.018]
Random + CRC	.20	[.096,.177]	[.174,.243]	[.000,.018]
ReCoverMem + CRC	.05	[.007,.042]	[.022,.072]	[.000,.018]
ReCoverMem + CRC	.10	[.042,.110]	[.141,.230]	[.005,.050]
ReCoverMem + CRC	.20	[.088,.216]	[.250,.389]	[.003,.043]

Mistral-7B-v0.3

Rule	α	FS 95% CI	Cov. 95% CI	Exc. 95% MC CI
Always Trust	–	[.343,.574]	[1.000,1.000]	–
Always Recover	–	[.000,.000]	[.000,.000]	–
Fixed-F1	–	[.237,.408]	[.583,.742]	–
TierMem router†	–	[.013,.089]	[.021,.120]	–
Empirical-risk	.05	[.017,.071]	[.036,.093]	[.224,.353]
Empirical-risk	.10	[.046,.127]	[.118,.238]	[.312,.451]
Empirical-risk	.20	[.137,.257]	[.325,.465]	[.129,.240]
Random + CRC	.05	[.007,.039]	[.037,.086]	[.000,.018]
Random + CRC	.10	[.036,.081]	[.116,.167]	[.000,.028]
Random + CRC	.20	[.104,.215]	[.287,.384]	[.000,.028]
ReCoverMem + CRC	.05	[.000,.025]	[.006,.034]	[.000,.018]
ReCoverMem + CRC	.10	[.033,.097]	[.074,.155]	[.003,.043]
ReCoverMem + CRC	.20	[.105,.224]	[.270,.415]	[.008,.057]

Gemma-3-12B

Rule	α	FS 95% CI	Cov. 95% CI	Exc. 95% MC CI
Always Trust	–	[.407,.615]	[1.000,1.000]	–
Always Recover	–	[.000,.000]	[.000,.000]	–
Fixed-F1	–	[.349,.527]	[.846,.963]	–
TierMem router†	–	[.015,.072]	[.021,.120]	–
Empirical-risk	.05	[.020,.085]	[.078,.187]	[.341,.482]
Empirical-risk	.10	[.049,.135]	[.149,.312]	[.265,.400]
Empirical-risk	.20	[.130,.250]	[.363,.545]	[.219,.348]
Random + CRC	.05	[.017,.059]	[.037,.086]	[.000,.018]
Random + CRC	.10	[.046,.092]	[.106,.160]	[.000,.028]
Random + CRC	.20	[.120,.218]	[.280,.374]	[.003,.043]
ReCoverMem + CRC	.05	[.005,.044]	[.029,.102]	[.000,.018]
ReCoverMem + CRC	.10	[.043,.114]	[.120,.263]	[.003,.043]
ReCoverMem + CRC	.20	[.120,.236]	[.340,.517]	[.017,.077]

Qwen3.8-27B

Rule	α	FS 95% CI	Cov. 95% CI	Exc. 95% MC CI
Always Trust	–	[.410,.677]	[1.000,1.000]	–
Always Recover	–	[.000,.000]	[.000,.000]	–
Fixed-F1	–	[.376,.624]	[.856,.963]	–
TierMem router†	–	[.019,.100]	[.021,.120]	–
Empirical-risk	.05	[.008,.038]	[.027,.078]	[.228,.358]
Empirical-risk	.10	[.045,.109]	[.160,.242]	[.178,.300]
Empirical-risk	.20	[.132,.300]	[.371,.551]	[.549,.688]
Random + CRC	.05	[.012,.046]	[.028,.074]	[.000,.018]
Random + CRC	.10	[.035,.081]	[.094,.149]	[.000,.018]
Random + CRC	.20	[.085,.173]	[.201,.272]	[.000,.018]
ReCoverMem + CRC	.05	[.008,.038]	[.027,.078]	[.000,.018]
ReCoverMem + CRC	.10	[.034,.088]	[.129,.193]	[.000,.018]
ReCoverMem + CRC	.20	[.117,.264]	[.333,.495]	[.129,.240]

†TierMem uses the frozen zero-shot routing decisions joined to each backbone's own memory-sufficiency labels.

C Scorers, Configurations, and Reproducibility

C.1 Memory-Sufficiency Scorer Details

Primary candidate-based scorer. The primary scorer is the logistic-regression model

$$s_t = \sigma\big(w^\top \phi(x_t, E_t, \tilde{a}_t) + b\big),$$

where σ denotes the logistic function. Its configuration-specific feature vector $\phi(x_t, E_t, \tilde{a}_t)$ summarizes evidence coverage and retrieval signals, support for the proposed memory-route action, and decision progress. The candidate $\tilde{a}_t$ is generated before routing but is not executed at this stage. Neither $R_{\mathrm{mem},t}$ nor $R_{\mathrm{rec},t}$ is an input feature.

Feature selection, fitted preprocessing parameters, and scorer parameters are determined using only the designated predictor-training data. The resulting scorer and preprocessing are frozen before calibration and applied unchanged to calibration and final-test inputs. Calibration labels select the routing threshold; they are not used to fit or select the scorer. Final-test labels are used only for evaluation.

Table A6: **PersonaMem-v2 uncertainty intervals.** FS/Cov. are pointwise 95% percentile bootstrap intervals over the 24 final-test personas ($B = 1000$, seed 13), with the scorer and calibrated threshold held fixed. Exc. entries are pointwise exact 95% Clopper–Pearson Monte Carlo intervals for the $B = 200$ archived split-pool replicates. Point estimates are reported in Table 1; only intervals are repeated here.

Llama-3.1-8B

Rule	α	FS 95% CI	Cov. 95% CI	Exc. 95% MC CI
Always Trust	–	[.676,.759]	[1.000,1.000]	–
Always Recover	–	[.000,.000]	[.000,.000]	–
Fixed-F1	–	[.638,.730]	[.928,.986]	–
TierMem router†	–	[.014,.053]	[.028,.077]	–
Empirical-risk	.05	[.028,.108]	[.045,.149]	[.513,.654]
Empirical-risk	.10	[.059,.154]	[.090,.223]	[.468,.611]
Empirical-risk	.20	[.104,.248]	[.153,.353]	[.399,.542]
Random + CRC	.05	[.000,.010]	[.000,.010]	[.000,.018]
Random + CRC	.10	[.031,.069]	[.031,.076]	[.017,.077]
Random + CRC	.20	[.149,.244]	[.191,.294]	[.104,.207]
ReCoverMem + CRC	.05	[.007,.031]	[.007,.031]	[.000,.028]
ReCoverMem + CRC	.10	[.031,.111]	[.049,.160]	[.083,.179]
ReCoverMem + CRC	.20	[.083,.212]	[.123,.306]	[.205,.332]

Mistral-7B-v0.3

Rule	α	FS 95% CI	Cov. 95% CI	Exc. 95% MC CI
Always Trust	–	[.648,.719]	[1.000,1.000]	–
Always Recover	–	[.000,.000]	[.000,.000]	–
Fixed-F1	–	[.632,.708]	[.965,.997]	–
TierMem router†	–	[.014,.050]	[.028,.077]	–
Empirical-risk	.05	[.028,.091]	[.042,.134]	[.351,.492]
Empirical-risk	.10	[.070,.165]	[.101,.232]	[.360,.502]
Empirical-risk	.20	[.174,.313]	[.250,.437]	[.404,.547]
Random + CRC	.05	[.000,.010]	[.000,.010]	[.000,.018]
Random + CRC	.10	[.021,.066]	[.035,.087]	[.035,.109]
Random + CRC	.20	[.136,.222]	[.207,.303]	[.062,.150]
ReCoverMem + CRC	.05	[.000,.000]	[.000,.010]	[.000,.018]
ReCoverMem + CRC	.10	[.032,.102]	[.047,.154]	[.024,.090]
ReCoverMem + CRC	.20	[.159,.292]	[.211,.391]	[.138,.251]

Gemma-3-12B-it

Rule	α	FS 95% CI	Cov. 95% CI	Exc. 95% MC CI
Always Trust	–	[.586,.703]	[1.000,1.000]	–
Always Recover	–	[.000,.000]	[.000,.000]	–
Fixed-F1	–	[.586,.703]	[1.000,1.000]	–
TierMem router†	–	[.014,.053]	[.028,.077]	–
Empirical-risk	.05	[.043,.130]	[.111,.195]	[.443,.586]
Empirical-risk	.10	[.089,.184]	[.185,.280]	[.434,.576]
Empirical-risk	.20	[.183,.308]	[.305,.433]	[.439,.581]
Random + CRC	.05	[.000,.010]	[.000,.010]	[.000,.018]
Random + CRC	.10	[.021,.069]	[.035,.087]	[.043,.121]
Random + CRC	.20	[.133,.217]	[.242,.334]	[.129,.240]
ReCoverMem + CRC	.05	[.007,.038]	[.014,.045]	[.000,.018]
ReCoverMem + CRC	.10	[.057,.148]	[.122,.220]	[.017,.077]
ReCoverMem + CRC	.20	[.156,.257]	[.260,.378]	[.142,.257]

Qwen3.8-27B

Rule	α	FS 95% CI	Cov. 95% CI	Exc. 95% MC CI
Always Trust	–	[.564,.669]	[1.000,1.000]	–
Always Recover	–	[.000,.000]	[.000,.000]	–
Fixed-F1	–	[.505,.625]	[.861,.958]	–
TierMem router†	–	[.008,.039]	[.028,.077]	–
Empirical-risk	.05	[.031,.118]	[.090,.201]	[.498,.640]
Empirical-risk	.10	[.085,.181]	[.165,.292]	[.498,.640]
Empirical-risk	.20	[.167,.301]	[.316,.493]	[.498,.640]
Random + CRC	.05	[.000,.010]	[.000,.010]	[.000,.018]
Random + CRC	.10	[.014,.045]	[.031,.076]	[.011,.064]
Random + CRC	.20	[.122,.203]	[.220,.309]	[.050,.133]
ReCoverMem + CRC	.05	[.000,.021]	[.000,.021]	[.000,.028]
ReCoverMem + CRC	.10	[.031,.118]	[.090,.201]	[.035,.109]
ReCoverMem + CRC	.20	[.129,.259]	[.261,.424]	[.129,.240]

†TierMem uses the frozen zero-shot routing decisions joined to each backbone's own memory-sufficiency labels.

Feature definitions and availability. Table A9 lists the ten fields in the primary candidate-based feature schema. Candidate dependence indicates whether computing an informative feature value requires the generated memory-route candidate; it does not imply an additional generation call beyond that candidate. Configuration-specific availability and preprocessing determine which fields are used as predictor inputs.

The archived Retail schema includes an `action_confidence` field, but candidate token log-probabilities were not recorded. Consequently, the field takes its fallback value of 0.5 for all 388 predictor-training, 424 calibration, and 439 final-test decisions and therefore provides no varying confidence signal in this configuration. Its treatment during preprocessing is specified in Table A10. The auxiliary LR-extra analysis adds a candidate log-probability signal, as described in Appendix E.2.

Preprocessing and fitting. Table A10 reports the fitting configuration for the Retail/Llama primary candidate-based scorer. Preprocessing specifies feature scaling, missing-value handling, and the treatment of constant or unavailable fields. Any fitted preprocessing statistics are estimated using predictor-training data only. Constancy on the predictor-training split alone does not establish constancy on calibration or final-test inputs.

Scorer-training weights are distinct from the unit-equal loss aggregation used by CRC. Training weights determine how individual examples contribute to fitting, whereas CRC averages decision losses within each unit and then weights units equally.

Configuration-specific fitting and transfer. Scorer configurations distinguish workloads, downstream backbones, host memories, and candidate-based or candidate-free variants. The Retail/Llama settings in Table A10 do not imply identical settings across configurations.

Table A7: **PrefEval uncertainty intervals under strict routing.** Routing uses TRUST iff $s_t > \tau$. FS/Cov. are pointwise 95% percentile bootstrap intervals over the 547 final-test sources ($B = 1000$, seed 20260920), with the scorer and calibrated threshold held fixed. Exc. entries are pointwise exact 95% Clopper–Pearson Monte Carlo intervals over $B = 200$ disjoint 200/200 splits of the frozen 400-source calibration pool. Point estimates are reported in Table 1; only intervals are repeated here.

Llama-3.1-8B

Rule	α	FS 95% CI	Cov. 95% CI	Exc. 95% MC CI
Always Trust	–	[.508,.590]	[1.000,1.000]	–
Always Recover	–	[.000,.000]	[.000,.000]	–
Fixed-F1	–	[.364,.444]	[.755,.823]	–
TierMem router†	–	[.005,.026]	[.009,.033]	–
Empirical-risk	.05	[.044,.084]	[.102,.163]	[.404,.547]
Empirical-risk	.10	[.071,.119]	[.170,.241]	[.458,.601]
Empirical-risk	.20	[.148,.210]	[.347,.430]	[.439,.581]
Random + CRC	.05	[.037,.075]	[.077,.126]	[.336,.477]
Random + CRC	.10	[.088,.137]	[.159,.223]	[.409,.552]
Random + CRC	.20	[.177,.243]	[.333,.419]	[.429,.571]
ReCoverMem + CRC	.05	[.035,.073]	[.095,.154]	[.322,.461]
ReCoverMem + CRC	.10	[.069,.119]	[.166,.238]	[.394,.537]
ReCoverMem + CRC	.20	[.144,.207]	[.336,.417]	[.380,.522]

Mistral-7B-v0.3

Rule	α	FS 95% CI	Cov. 95% CI	Exc. 95% MC CI
Always Trust	–	[.563,.640]	[1.000,1.000]	–
Always Recover	–	[.000,.000]	[.000,.000]	–
Fixed-F1	–	[.300,.373]	[.605,.686]	–
TierMem router†	–	[.005,.024]	[.009,.033]	–
Empirical-risk	.05	[.033,.071]	[.091,.148]	[.439,.581]
Empirical-risk	.10	[.073,.122]	[.186,.252]	[.404,.547]
Empirical-risk	.20	[.170,.232]	[.377,.452]	[.448,.591]
Random + CRC	.05	[.042,.080]	[.077,.128]	[.332,.471]
Random + CRC	.10	[.088,.139]	[.150,.214]	[.360,.502]
Random + CRC	.20	[.192,.260]	[.333,.419]	[.356,.497]
ReCoverMem + CRC	.05	[.033,.071]	[.088,.143]	[.327,.466]
ReCoverMem + CRC	.10	[.073,.122]	[.185,.250]	[.346,.487]
ReCoverMem + CRC	.20	[.161,.225]	[.356,.435]	[.399,.542]

Gemma-3-12B-it

Rule	α	FS 95% CI	Cov. 95% CI	Exc. 95% MC CI
Always Trust	–	[.495,.576]	[1.000,1.000]	–
Always Recover	–	[.000,.000]	[.000,.000]	–
Fixed-F1	–	[.446,.527]	[.890,.936]	–
TierMem router†	–	[.004,.022]	[.009,.033]	–
Empirical-risk	.05	[.046,.086]	[.093,.148]	[.468,.611]
Empirical-risk	.10	[.080,.132]	[.168,.238]	[.458,.601]
Empirical-risk	.20	[.132,.196]	[.278,.358]	[.453,.596]
Random + CRC	.05	[.051,.093]	[.095,.150]	[.308,.446]
Random + CRC	.10	[.102,.155]	[.196,.267]	[.351,.492]
Random + CRC	.20	[.192,.260]	[.378,.464]	[.365,.507]
ReCoverMem + CRC	.05	[.042,.082]	[.091,.144]	[.394,.537]
ReCoverMem + CRC	.10	[.080,.128]	[.159,.225]	[.404,.547]
ReCoverMem + CRC	.20	[.132,.196]	[.278,.358]	[.419,.561]

Qwen3.8-27B

Rule	α	FS 95% CI	Cov. 95% CI	Exc. 95% MC CI
Always Trust	–	[.598,.675]	[1.000,1.000]	–
Always Recover	–	[.000,.000]	[.000,.000]	–
Fixed-F1	–	[.594,.673]	[.982,.998]	–
TierMem router†	–	[.007,.027]	[.009,.033]	–
Empirical-risk	.05	[.035,.073]	[.079,.133]	[.434,.576]
Empirical-risk	.10	[.090,.144]	[.185,.250]	[.424,.566]
Empirical-risk	.20	[.161,.229]	[.309,.388]	[.424,.566]
Random + CRC	.05	[.035,.075]	[.077,.128]	[.284,.420]
Random + CRC	.10	[.090,.141]	[.159,.223]	[.356,.497]
Random + CRC	.20	[.196,.265]	[.336,.422]	[.370,.512]
ReCoverMem + CRC	.05	[.033,.069]	[.079,.132]	[.365,.507]
ReCoverMem + CRC	.10	[.090,.144]	[.185,.250]	[.380,.522]
ReCoverMem + CRC	.20	[.161,.229]	[.307,.386]	[.385,.527]

†TierMem uses the frozen zero-shot routing decisions joined to each backbone's own memory-sufficiency labels.

Airline reuses the Retail-trained scorer and selects a separate threshold on Airline calibration episodes. The Retail host-memory comparisons fit and calibrate a separate scorer for each host. NarrativeXL fits candidate-free scorers separately for each downstream backbone using 50 predictor-training books, followed by calibration on 50 separate books and evaluation on 50 held-out test books containing 600 questions in total.

Candidate-free variant. The candidate-free scorer computes

$$s_t^{\mathrm{cf}} = s_{\mathrm{cf}}(x_t, E_t)$$

before candidate generation. It excludes candidate-answer content, candidate log-probabilities, and any other quantity requiring generation of $\tilde{a}_t$. Each evaluated candidate-free configuration has its own fitted scorer and separately calibrated threshold; it does not reuse the candidate-based threshold. Fitting uses predictor-training data, and CRC selects the threshold using candidate-free scores and memory-route labels on the calibration split.

The false-safe definition, calibration units, strict routing rule, and target risk remain unchanged. The variant changes the scorer inputs and permits routing to RECOVER without first generating a memory-route candidate. Only the selected branch requires an answer-generation call. Calls used solely to obtain offline paired branch outcomes are excluded from required deployment calls. Candidate-free routing can therefore change both the selected operating point and the executed-call cost.

Auxiliary scorer analyses. Auxiliary scorers assess ranking quality and sensitivity to feature design. Each fitted variant uses predictor-training data for fitting and a separate CRC threshold selected on calibration data. These variants do not replace the frozen primary scorer in the corresponding main experiment unless explicitly identified.

Table A8: **Paired differences at $\alpha = .10$.** 95% paired-bootstrap intervals for $\Delta = ReCoverMem + \text{CRC} - \text{comparator}$, in percentage points. *Excludes zero. †Exact zero: the two policies select the same threshold, or thresholds that differ only on decisions with sufficient memory, so the per-unit difference is zero in every resample.

Workload	Backbone	Comparator	ΔFS 95% CI	ΔCov. 95% CI
Retail	Llama-3.1-8B	Random + CRC	[-3.40,+4.10]	[+2.39,+11.60]*
Retail	Llama-3.1-8B	Empirical-risk	[-5.49,-0.42]*	[-6.25,-1.04]*
Retail	Mistral-7B-v0.3	Random + CRC	[-2.71,+3.75]	[-7.29,+1.67]
Retail	Mistral-7B-v0.3	Empirical-risk	[-3.96,-0.62]*	[-9.53,-3.69]*
Retail	Gemma-3-12B-it	Random + CRC	[-2.78,+4.86]	[-1.46,+13.23]
Retail	Gemma-3-12B-it	Empirical-risk	[-3.54,-0.21]*	[-7.29,-1.46]*
Retail	Qwen3.8-27B	Random + CRC	[-2.50,+3.30]	[-0.14,+8.20]
Retail	Qwen3.8-27B	Empirical-risk	[-3.13,-0.42]*	[-6.09,-2.19]*
PersonaMem-v2	Llama-3.1-8B	Random + CRC	[-1.74,+6.25]	[-0.69,+11.11]
PersonaMem-v2	Llama-3.1-8B	Empirical-risk	[-6.00,-1.74]*	[-8.11,-2.78]*
PersonaMem-v2	Mistral-7B-v0.3	Random + CRC	[-1.71,+6.41]	[-2.37,+10.45]
PersonaMem-v2	Mistral-7B-v0.3	Empirical-risk	[-8.05,-2.78]*	[-9.75,-3.85]*
PersonaMem-v2	Gemma-3-12B-it	Random + CRC	[-0.69,+11.75]	[+4.61,+17.49]*
PersonaMem-v2	Gemma-3-12B-it	Empirical-risk	[-5.56,-1.74]*	[-9.37,-3.46]*
PersonaMem-v2	Qwen3.8-27B	Random + CRC	[+0.00,+9.12]	[+3.13,+15.41]*
PersonaMem-v2	Qwen3.8-27B	Empirical-risk	[-8.74,-3.25]*	[-11.14,-5.62]*
PrefEval	Llama-3.1-8B	Random + CRC	[-5.10,+1.60]	[-3.80,+5.90]
PrefEval	Llama-3.1-8B	Empirical-risk	[-0.50,+0.00]	[-0.90,+0.00]
PrefEval	Mistral-7B-v0.3	Random + CRC	[-4.90,+1.80]	[-1.60,+8.00]
PrefEval	Mistral-7B-v0.3	Empirical-risk	[+0.00,+0.00]†	[-0.50,+0.00]
PrefEval	Gemma-3-12B-it	Random + CRC	[-6.20,+0.70]	[-9.00,+0.90]
PrefEval	Gemma-3-12B-it	Empirical-risk	[-0.50,+0.00]	[-2.00,-0.40]*
PrefEval	Qwen3.8-27B	Random + CRC	[-3.50,+3.70]	[-2.20,+7.50]
PrefEval	Qwen3.8-27B	Empirical-risk	[+0.00,+0.00]†	[+0.00,+0.00]†
NarrativeXL	Llama-3.1-8B	Random + CRC	[-3.50,+2.34]	[-3.17,+4.83]
NarrativeXL	Gemma-3-12B-it	Random + CRC	[-3.83,+1.00]	[-1.17,+7.17]
NarrativeXL	Qwen3.8-27B	Random + CRC	[-3.83,+1.67]	[-3.17,+4.50]

Ranking performance and calibrated false-safe risk and trust coverage are reported separately, since better ranking alone does not guarantee greater coverage at a particular calibrated operating point. Auxiliary configurations and results are reported in Appendix E.2.

C.2 Workload and Baseline Configurations

Cross-backbone evaluation. The primary fixed-state workloads are τ^3-Bench Retail, PersonaMem-v2, and PrefEval. They use Mem0 with Llama-3.1-8B-Instruct as the memory writer. Downstream backbones are Llama-3.1-8B-Instruct, Mistral-7B-Instruct-v0.3, Gemma-3-12B-it, and Qwen3.8-27B. Within each workload, retrieved memory evidence, recovery evidence, checkpointed states, and split assignments are frozen across downstream backbones. Branch outcomes and fitted scorers are backbone-specific. Scorer fitting uses only predictor-training data, and threshold selection uses calibration data without access to final-test outcomes.

Boundary configurations. Airline and ALFWorld provide low- and high-sufficiency configurations, respectively. Airline uses the Retail-trained scorer with calibration on its own episode-level units. These experiments characterize fixed-state operating regimes; sequential deployment outcomes are reported separately as on-policy diagnostics.

NarrativeXL. NarrativeXL evaluates candidate-free routing under a served 32K context budget. Mem0 memories, retrieved evidence, and book splits are fixed across downstream backbones. The split contains 50 predictor-training, 50 calibration, and 50 final-test books. The final-test split contains 12 questions per book, for 600 questions in total. Candidate-free scorers are fitted separately for each evaluated backbone, with thresholds selected on the corresponding calibration scores. The scorers use question and retrieved-evidence features without generating a candidate answer, so deployment requires answer generation only for the selected branch.

Host-memory portability. Retail host-memory comparisons use Mem0, Rolling Summary, A-Mem, and Three-Layer Memory with a fixed task split and Llama-3.1-8B-Instruct downstream backbone. Three-Layer Memory uses the frozen bounded v2 implementation. Each host receives its own

Table A9: **Memory-sufficiency scorer features.** The table records exact feature keys, computations, candidate dependence, and configuration-specific use. Schema membership does not imply that a field provides a varying predictor input in every configuration.

Feature key	Computation	Candidate-dependent?	Configurations
`retrieval_coverage`	Fraction of query terms appearing in retrieved memory evidence: $\lvert Q_{\text{term}} \cap E_{\text{term}}\rvert / \lvert Q_{\text{term}}\rvert$; an empty query-term set maps to 1.	No	Primary candidate-based schema.
`entity_coverage`	Fraction of query entities appearing in retrieved memory evidence: $\lvert Q_{\text{ent}} \cap E_{\text{ent}}\rvert / \lvert Q_{\text{ent}}\rvert$; an empty query-entity set maps to 1.	No	Primary candidate-based schema; constant at 1.0 in the audited ALFWorld predictor data.
`action_arg_support`	Fraction of candidate-action argument literals found among retrieved evidence terms or entities; an empty argument-literal set maps to 1.	Yes	Primary candidate-based schema; constant at 1.0 in audited PersonaMem-v2 and PrefEval predictor data.
`top_similarity`	Largest numeric retrieval similarity score, clipped to $[0, 1]$; 0 if no numeric retrieval score is available.	No	Primary candidate-based schema.
`similarity_margin`	Top minus second-highest retrieval similarity, clipped to $[0, 1]$; equals the clipped top score when only one score is available and 0 when none is available.	No	Primary candidate-based schema.
`support_count_norm`	$\text{clip}_{[0,1]}(\log(1 + \lvert E\rvert)/\log 9)$, where $\lvert E\rvert$ is the number of retrieved evidence items; the value saturates at eight items.	No	Primary candidate-based schema; constant at 1.0 in audited PersonaMem-v2 and PrefEval predictor data.
`evidence_utilization`	$\text{clip}_{[0,1]}(B_{\text{used}}/\max(1, B_{\text{mem}}))$, using retrieved-evidence tokens relative to the evidence budget.	No	Primary candidate-based schema.
`conflict_density_inv`	One minus the fraction of entity-sharing evidence-item pairs with mismatched negation status; equals 1 when no eligible pair exists.	No	Primary candidate-based schema; constant at 1.0 in the audited PrefEval predictor data and ALFWorld.
`action_confidence`	$\text{clip}_{[0,1]}(\exp(\overline{\log p}))$ from the candidate's mean token log-probability; missing log-probability maps to 0.5.	Yes	Primary candidate-based schema; constant at 0.5 in archived Retail, Airline, and A-Mem runs.
`step_progress`	$\text{clip}_{[0,1]}(t/\max(1, T_{\max}))$, using the current step index and maximum step count.	No	Primary candidate-based schema; constant at 0 in audited PersonaMem-v2 and PrefEval data.

memory-route labels, fitted scorer, and calibrated threshold. The comparison evaluates portability with host-specific fitting, rather than transfer of one frozen scorer. Hosts may induce different controlled-decision populations even when they share initial tasks and split assignments.

Routing and baseline interface. Threshold-based fixed-state policies use

$$\textsc{Trust} \iff s_t > \tau, \qquad \textsc{Recover} \iff s_t \leq \tau.$$

Table A10: **Retail/Llama scorer fitting configuration.** Settings refer to the primary candidate-based logistic-regression scorer. Training counts distinguish episode-level units from individual controlled decisions.

Setting	Value
Workload / backbone	Retail / Llama-3.1-8B-Instruct
Scorer variant	Primary candidate-based
Model family	L2-regularized logistic regression
Predictor-training units / decisions	34 episodes / 388 decisions
Positive / negative training labels	126 / 262
Model-input feature keys	`retrieval_coverage`, `entity_coverage`, `action_arg_support`, `top_similarity`, `similarity_margin`, `support_count_norm`, `evidence_utilization`, `conflict_density_inv`, `action_confidence`, `step_progress`
Constant / unavailable fields	`action_confidence`=0.5 is the only constant predictor-training field; all other inputs vary
Treatment of constant / unavailable fields	Retained in the 10-feature schema. Zero-variance scale is set to 1, so `action_confidence` standardizes to 0 and has fitted coefficient 0
Feature scaling	$z = (x - \mu)/\sigma$, using predictor-training column mean and population SD (`ddof`=0); zero SD is replaced by 1; no clipping
Missing-value handling	No generic imputation. Missing keys or invalid numerical values fail; unavailable candidate log-probability is encoded as `action_confidence`=0.5
Regularization strength	$C = 1.0$
Maximum iterations	500
Class weights	`balanced`
Training-sample weights	None
Fitting random seed	`random_state`=13
Hyperparameter-selection procedure	Fixed final-fit configuration carried over from the pre-existing implementation; no search or cross-validation in the primary fit
Feature schema	`recovermem-features-v1`

Ties therefore recover. The primary candidate-based scorer uses a memory-route candidate generated before routing; the candidate-free variant excludes candidate-dependent features.

Reference and threshold-selection baselines. ALWAYS TRUST selects the memory route at every controlled decision, whereas ALWAYS RECOVER always selects recovery. Empirical-risk uses the same frozen learned score as ReCoverMem but omits the CRC finite-sample correction. Random + CRC applies CRC to a frozen label-independent random ranking.

Fixed-F1 uses the same frozen learned score as ReCoverMem and selects its threshold by maximizing decision-pooled F1 on the calibration split, with $R_{\text{mem}} = 1$ as the positive class and TRUST as the predicted positive. Candidate thresholds comprise the distinct calibration scores and $-\infty, +\infty$. Equal-F1 ties are resolved in favor of the larger threshold; undefined precision, recall, or F1 values are assigned zero. Final-test labels are not used for threshold selection. F1 is computed over pooled decisions, whereas reported FS and coverage use unit-equal weighting. These weighting schemes coincide for expanded PrefEval, which has one controlled decision per source.

Across Retail, PersonaMem-v2, and expanded PrefEval, Fixed-F1 thresholds are selected and evaluated using the strict routing rule $s_t > \tau$, with ties routed to RECOVER. The same strict rule is used by the threshold-based policies throughout our final evaluation.

Zero-shot TierMem and parsing fallback. The zero-shot TierMem router uses the same TRUST/RECOVER interface while keeping the host memory and recovery operator fixed. For expanded PrefEval, the router's output labels `S` and `R` map to TRUST and RECOVER, respectively. Routing uses temperature 0 and seed 13, with one router call per source and no retries.

The reported expanded PrefEval results retain the native fallback for outputs that fail structured parsing. Such outputs use the recorded `native_fallback_decision`, with `S` mapped to TRUST and `R` to RECOVER; parse failure does not automatically trigger recovery. Among the 547 final-test sources, 10 outputs fail structured parsing: the native fallback routes nine to TRUST and one to RECOVER. All 547 sources remain in the evaluation denominator. The fallback requires no additional model call and uses the frozen memory-route outcome when computing false-safe loss.

The audited backbone comparisons share the same TierMem routing decisions on equivalent routing inputs while using backbone-specific memory-route labels. Their identical coverage therefore reflects shared routing decisions, whereas their false-safe risks can differ.

Auxiliary scorer analyses are reported in Appendix E.2.

Operational success labels. The memory-route label R_{mem} is defined by a workload-specific criterion fixed before evaluation. Binary evaluators provide the label directly, whereas utility-based evaluators use a predefined threshold. For τ^3-Bench Retail, the memory route is labeled successful when

$$u_{\text{mem}} \geq .5,$$

which indicates that the proposed tool name matches an outstanding gold action. This criterion does not certify correct tool arguments or complete task success.

For every workload below, R_{rec} applies the same workload-specific success criterion to the recovery branch from the shared checkpoint; only the evidence source and recovery operator differ.

- **PersonaMem-v2:** The frozen PersonaMem-v2 multiple-choice evaluator parses the branch answer and compares it with the designated correct option. The utility is binary, with $u = 1$ for a correct parsed choice and $u = 0$ otherwise, and $R = 1[u \geq .5]$. An output that does not yield the correct parsed option is therefore labeled unsuccessful.
- **PrefEval:** The released programmatic multiple-choice evaluator assigns $u = 1$ iff the parsed choice equals the shuffled gold option and $u = 0$ otherwise, with $R = 1[u \geq .5]$. An invalid or unparsable output has no correct parsed choice and is therefore labeled unsuccessful.
- **τ^3-Bench Airline:** Airline uses the same τ^3 action-level evaluator as Retail, with $R = 1[u \geq .5]$. Success indicates that the proposed tool name matches an outstanding gold action; it does not require correct tool arguments or final task completion.
- **ALFWorld:** The frozen evaluator uses a binary branch-segment utility with threshold $\gamma = 1$. A branch receives $u = 1$ when the bounded branch segment satisfies the evaluator's progress criterion and $u = 0$ otherwise, so $R = 1[u \geq 1]$. This is a controlled-boundary branch-success label rather than final episode success.
- **NarrativeXL:** The branch output is parsed as a multiple-choice option and $R = 1$ iff the predicted option equals the gold option. Invalid or unparsable outputs are labeled unsuccessful.

Paired fixed-state evaluation. At each controlled checkpoint, the memory and recovery branches are evaluated from the same underlying state. The downstream backbone, decoding configuration, evaluator, and shared checkpoint state are held fixed, while the two routes receive their respective memory or historical evidence. The resulting pair $(R_{\text{mem}}, R_{\text{rec}})$ supports fixed-state routed-success and repair/harm diagnostics. These paired outcomes do not by themselves identify the mechanism that caused a memory-route failure.

Host and evidence routes. The primary controller leaves Mem0's write, update, and retrieval mechanisms unchanged. The memory route receives retrieved host evidence, while recovery obtains historical evidence from the frozen trajectory. The scorer does not receive recovery outcomes or unrestricted raw-history evidence.

Empty units and protocol scope. The empty-unit convention is fixed before calibration and applied consistently to the corresponding target population. Retail excludes two empty calibration episodes, and the ALFWorld counts distinguish designated from non-empty units in Table A1. The

strict rule applies to the recalculated fixed-state results. Historical on-policy rollouts are reported separately and are not represented as newly recomputed fixed-state experiments.

C.3 FROZEN RETAIL IMPLEMENTATION

The following implementation details describe the archived Retail/Llama configuration; they are not a universal serving specification for every backbone.

Common-state construction. Both branches receive the same common state x_t; only their evidence source differs. We exclude hidden simulator state from x_t and retain only information already observable to the native agent.

An observation ledger records actions already executed and identifiers that the agent has legitimately observed. Tool outputs themselves remain in the trajectory and are not copied into the common state. Action parameters that cannot be traced to the observed corpus are replaced by `<unobserved>`.

The recorded checks detect no declarative state-leakage violations or hidden-value leaks in the retained predictor-training and final-test states.

Serving determinism. Paired evaluation uses temperature 0 and requires repeated execution from the same state to be stable at the action level.

The archived Retail collection used vLLM v0.18.0 with automatic prefix caching enabled and `enforce_eager=False`, a 32,768-token context window, and `gpu_memory_utilization=0.85`. The later `--no-enable-prefix-caching` and `--enforce-eager` flags were introduced in a stricter determinism-safe serving configuration and were not used for the archived Retail collection.

Frozen configuration. The τ^3-Bench Retail split is frozen at the episode level. The initial partition contains 34 predictor-training episodes, 40 calibration episodes, and 40 final-test episodes. Two calibration episodes contain no controlled decisions for which native utility is defined. CRC therefore operates on the 38 non-empty calibration episodes, giving an effective

$$n_{\mathrm{cal}} = 38.$$

The predictor is trained on 34 episodes containing 388 controlled decisions, with 126 positive and 262 negative R_{mem} labels. The fitted model is frozen before calibration.

Component	Frozen value
τ^3-Bench commit	`a2c024725189473d2d7cea3a5cfdbcc67478e41f`
Mem0 commit	`39bc02330563764e7d4465f1ecff5f002d94da1a`
Agent / memory LLM	Llama-3.1-8B-Instruct, local vLLM
Embedding model	all-MiniLM-L6-v2, 384-d
Context window	32768 tokens
$B_{\mathrm{mem}}/B_{\mathrm{rec}}$	1024 / 1024 tokens
Operational success threshold γ	0.5
Split seed	13
Random-score seed	13
Exc. resamples	200

On the final Retail test set, the memory-sufficiency predictor obtains AUROC 0.6439 and AUPRC 0.4417. The unit-equal empirical memory-sufficiency prevalence is

$$\widehat{\pi} = 0.3815,$$

giving the sanity check

$$\widehat{\mathrm{FS}}(\textsc{Always Trust}) = 1 - \widehat{\pi} = 0.6185.$$

Retail evidence budget. The configured decision-time evidence cap is 1,024 tokens for each route. This is distinct from the 32,768-token context window and from persistent memory-store capacity. In the archived budget audit, native Mem0 evidence reached at most 926 serialized tokens, and the recovery cap was binding on 32.2% of the audited states. These observations concern the evidence-budget audit, not the final 439-state context-occupancy analysis in Appendix G.3.

D METRICS, COST ACCOUNTING, AND UNCERTAINTY

D.1 UNIT-EQUAL METRICS

For non-empty unit i, let $d_i = |\mathcal{T}_i|$. Empirical FS and coverage first average over its d_i decisions and then over units:

$$\widehat{\mathrm{FS}} = \frac{1}{N}\sum_{i=1}^{N} L_i(\hat{\tau}), \qquad \widehat{\mathrm{Cov}} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{d_i}\sum_{t\in\mathcal{T}_i} \mathbb{1}[s_{i,t} > \hat{\tau}].$$

Retained empty units contribute zero by convention.

Conditional error among trusted decisions. When $\mathrm{Cov}(\delta) > 0$, define

$$\mathrm{TrustedError}(\delta) = \frac{\mathrm{FS}(\delta)}{\mathrm{Cov}(\delta)}.$$

This is the conditional memory-route error rate under the sampling scheme that first selects a unit with equal weight and then selects a decision uniformly within a non-empty unit, conditioning on TRUST. It is not generally the unweighted mean of per-unit trusted-error ratios. Retained empty units contribute no trusted event. TrustedError is undefined when coverage is zero.

Controlling marginal false-safe risk does not control TrustedError at the same target level. A policy can have low false-safe risk because it rarely trusts, even if a substantial fraction of its trusted decisions fail.

Any false-safe within a unit. Define

$$\mathrm{AnyFS}(\delta) = \Pr(\exists\, t \in \mathcal{T}_i : \mathrm{FS}_\delta(i,t) = 1)\,,$$

with no event for an empty unit. This quantity measures whether a unit contains at least one false-safe, whereas our calibrated loss measures the fraction of controlled decisions that are false-safe. The CRC guarantee for the latter does not provide an α-level bound on AnyFS.

Exceedance. Exceedance (**Exc.**) is the fraction of repeated calibration draws or calibration–test resamples whose empirical test FS exceeds the target α. The statistical unit, calibration pool, calibration size, and whether the test set is fixed or resampled are specified for each analysis.

Exc. is a finite-sample stability diagnostic. It is not the probability that the CRC theorem is violated, since the CRC guarantee concerns expected future-unit loss rather than requiring every realized empirical test set to satisfy $\widehat{\mathrm{FS}} \le \alpha$.

Fixed-state versus on-policy outcomes. In fixed-state experiments, routed success combines the frozen outcome of whichever branch the routing policy selects at each shared checkpoint. The paired memory and recovery outcomes separately characterize recovery repair and harm.

For sequential Retail and Airline deployment experiments, each policy executes its own actions and therefore induces its own subsequent states. Policies are paired by their initial task rather than by later visited states. These on-policy results are consequently reported separately from fixed-state CRC evaluation.

Static-query and on-policy cost. For static-query and on-policy analyses, normalized input cost is the unit-equal mean of per-unit input-token ratios to the corresponding frozen recovery reference. For unit i,

$$C_i^{\mathrm{norm}} = \frac{C_i^{\mathrm{policy,input}}}{C_i^{\mathrm{recovery\text{-}ref,input}}}, \tag{8}$$

and the reported normalized cost averages C_i^{norm} equally across units.

Candidate-generation and recovery inputs are included when required by the evaluated execution path. Completion tokens, host-memory maintenance, scorer training, calibration, and evaluation-only shadow calls are excluded unless explicitly stated otherwise. Absolute token counts are reported separately from normalized costs.

NarrativeXL cost. NarrativeXL reports a different cost convention: a ratio of book-equal mean input-plus-output token counts. Candidate-free execution and the choice of normalization are separate accounting decisions. Its normalized cost is

$$C_{\text{NXL}}^{\text{norm}} = \frac{\mathbb{E}_{\text{book}}[C_{\text{executed}}^{\text{input+output}}]}{\mathbb{E}_{\text{book}}[C_{\text{AR}}^{\text{input+output}}]}, \tag{9}$$

where the numerator counts only the answer-generation branch actually executed after routing.

Calls used solely to obtain offline paired outcomes, host-memory maintenance, scorer training, calibration, and evaluation-only shadow calls are excluded. Accordingly, NarrativeXL normalized costs are not numerically comparable to the input-only normalized costs reported for static-query or on-policy experiments.

Uncertainty. Uncertainty is estimated by resampling statistical units: episodes for episodic workloads, sources for source-grouped workloads, and books for NarrativeXL. Repeated decisions or questions within a unit remain grouped throughout resampling, and paired comparisons reuse the same sampled units.

Calibration-draw variability is reported separately from final-test sampling uncertainty. Bootstrap replicate counts and random seeds are specified with the corresponding analyses. Unless explicitly stated otherwise, confidence intervals quantify test-unit sampling uncertainty with the scorer and calibrated threshold held fixed.

Interpretation of trusted error. For Retail/Llama at the primary target, the rounded main-table values give $\text{TrustedError} \approx .075/.182 \approx .41$. This illustrates why marginal FS and error conditional on Trust must be reported as distinct quantities.

E Supplementary Risk and Scorer Analyses

E.1 Operating-Regime Diagnostics

Figure 3 in the main text compares actual coverage with an empirical inequality and a nominal-budget reference. On Airline/Llama, $\widehat{\pi} = .128$ and calibrated coverage is zero at $\alpha = .10$. The nominal-budget reference is .228; this does not require the selected operating point to have zero coverage. On ALFWorld/Qwen3-32B-AWQ, $\widehat{\pi} = .963$ and Always Trust has empirical FS .037. The calibrated policy has the same FS and coverage .886. These are fixed-state observations.

The oracle-label diagnostic uses memory-route outcomes as scores and is recalibrated. It is not deployable and is not an optimizer over all policies: binary scores and strict thresholding restrict its available operating points. Its coverage need not attain $\min(1, \widehat{\pi} + \alpha)$.

E.2 Scorer and Feature Ablations

The auxiliary Retail/Llama scorer study holds the fixed-state evaluation population constant. Feature and model selection use the 34 predictor-training episodes; episode-grouped five-fold cross-validation is used where selection is required. The primary scorer remains frozen.

Auxiliary features. LR-extra replays the frozen candidate-generation procedure to obtain mean candidate token log-probability, without executing the action or using outcome labels as inputs. Among three predeclared auxiliary signals, evidence token count is redundant with evidence utilization, and the specified NLI model cannot process 144 of 388 training pairs without the truncation prohibited by that protocol. Only the log-probability signal is retained. It improves recorded AUROC from .644 to .679 but does not redefine the primary scorer.

Table A11: **Auxiliary Retail/Llama ranking diagnostics.** AUROC and AUPRC are the held-out ranking statistics recorded in the scorer study. These metrics do not depend on the routing threshold.

Scorer	AUROC	AUPRC
Top similarity	.391	.306
ReCoverMem LR	.644	.442
LR-extra	.679	.464
GBDT	.670	.461
LLM self-judge	.580	.407

Self-judging. The Llama self-judge receives the pre-routing state and memory evidence and scores sufficiency using the normalized log-probabilities of two frozen answer tokens. It requires an additional LLM call per decision. Its recorded Retail ranking statistics appear in Table A11; they do not establish superiority in calibrated coverage or execution cost.

Available provenance features. The recorded feature ablations do not show a clear incremental gain from the available provenance features. ADD/UPDATE indicators have limited variation, and compression depth and consolidation lineage were not effectively tested. This limits conclusions to the measured features and configuration; it does not establish that compression provenance is generally uninformative.

F Efficiency and Deployment Details

F.1 NarrativeXL Risk–Cost–Utility Details

NarrativeXL evaluates candidate-free routing under a served 32K context budget. For each backbone, the scorer is fitted on 50 training books, calibrated on 50 disjoint books, and evaluated on 50 held-out books with 12 questions per book. Mem0 memories, retrieved evidence, and book splits are fixed across backbones. Routing occurs before answer generation, so deployment cost includes only the selected answer-generation branch in the reported token metric.

Paired uncertainty at the primary target. Table A12 reports paired book-bootstrap intervals at $\alpha = .10$. We use 1,000 paired book-bootstrap draws (seed 20260919), holding the fitted scorers and thresholds fixed. The intervals quantify test-book sampling uncertainty, not calibration uncertainty. Token savings are measured relative to Always Recover. The learned-versus-random contrasts use the same held-out books and preserve within-book dependence.

Table A12: **Paired NarrativeXL uncertainty at $\alpha = .10$.** Savings and ΔSuccess are relative to Always Recover. ΔCov. and $\Delta\text{Success}_{\text{R}-\text{Rand}}$ compare ReCoverMem with Random + CRC. Every bracketed interval is a paired 95% book-bootstrap interval with frozen scorers and thresholds; differences are method minus comparator.

Backbone	Saving (%) [95% CI]	ΔSuccess (pp) vs. AR	ΔCov. (pp) vs. Random	ΔSuccess (pp) vs. Random
Qwen3.8-27B	8.5 [6.5, 10.5]	-5.3 $[-7.3, -3.7]$	$+0.8$ $[-3.2, +4.5]$	-0.3 $[-2.3, +2.0]$
Llama-3.1-8B	11.1 [9.2, 13.5]	-7.7 $[-9.8, -5.7]$	$+0.7$ $[-3.2, +4.8]$	-2.5 $[-5.2, +0.5]$
Gemma-3-12B-it	13.6 [11.3, 16.3]	-3.0 $[-5.2, -1.0]$	$+3.0$ $[-1.2, +7.2]$	0.0 $[-2.7, +2.8]$

All three token-saving intervals lie above zero, whereas all learned-versus-random coverage intervals include zero. Thus, the evidence supports a measurable efficiency effect from candidate-free selective routing, while the incremental coverage benefit of learned ranking remains unresolved on the 50-book test set. The corresponding routed-success differences between ReCoverMem and Random + CRC are likewise not statistically resolved.

Risk-target trade-offs. Figure 4 reports the operating points across the evaluated risk targets. The figure separates false-safe risk and coverage from the associated token and routed-success changes relative to ALWAYS RECOVER.

A stricter target can select a milder cost–success trade-off. For example, at $\alpha = .05$, Gemma saves 3.9% of tokens (95% CI $[2.5, 5.3]$), while routed success changes by -0.5 percentage points (95% CI $[-1.5, 0.5]$). Across the three valid backbones and three risk targets, empirical FS is at or below the nominal target in eight of nine operating points; the exception is Qwen at $\alpha = .20$, with FS .2017.

Reference policies and cost accounting. At $\alpha = .10$, ALWAYS RECOVER success is $.790/.792/.780$ for Qwen/Llama/Gemma and has zero FS and coverage. ALWAYS TRUST success is $.298/.313/.462$, with normalized token cost $.272/.257/.252$. Normalized cost is the ratio of book-equal mean executed-branch input-plus-output tokens to the corresponding ALWAYS RECOVER mean. Memory maintenance and evaluation-only shadow calls are excluded. Mistral is excluded because it failed the frozen output-format gate.

The candidate-free scorers achieve held-out memory-sufficiency AUROC of .585–.616, consistent with modest discrimination between successful and unsuccessful memory routes. These results support executed-branch token savings with a measured success trade-off; the incremental learned-versus-random coverage advantage is not statistically resolved.

Table A13: **NarrativeXL reference policies.** AT denotes Always Trust and AR denotes Always Recover. AT coverage is one; AR FS and coverage are zero and its normalized cost is one. Costs use the ratio of book-equal mean executed-branch input-plus-output tokens.

Backbone	AT FS	AT success	AT cost	AR success
Qwen3.8-27B	.702	.298	.272	.790
Llama-3.1-8B	.687	.313	.257	.792
Gemma-3-12B-it	.538	.462	.252	.780

F.2 RECOVERY AND ON-POLICY DEPLOYMENT DETAILS

The main text distinguishes false-safe risk from recovery effectiveness and end-task utility. Here we report the corresponding fixed-state branch diagnostics and the complete recorded on-policy results.

Recovery repair and harm. On fixed-state Retail/Llama, recovery sometimes repairs an insufficient memory route but can also disrupt a successful one. These are conditional quantities with different denominators and therefore should not be subtracted as a net benefit measure.

Table A14: **Fixed-state recovery diagnostics on Retail/Llama.** Repair is conditioned on memory-route failure; harm is conditioned on memory-route success. Rates use unit-equal weighting. Because the denominators differ, the two percentages do not define a single net recovery effect.

Diagnostic	Conditional recovery outcome	Rate
Repair	$R_{\text{rec}} = 1 \mid R_{\text{mem}} = 0$	6.6%
Harm	$R_{\text{rec}} = 0 \mid R_{\text{mem}} = 1$	15.8%

These measurements describe both frozen branches at matched checkpoints. They do not measure only those recoveries selected by a particular routing policy. Thus, memory-sufficiency prediction and recovery effectiveness remain separate components of the deployed system.

Deployment cost and scope. The candidate-based Retail rollout retains the pre-routing generation overhead analyzed in Section 4.4. Its 455 routing-stage model calls comprise 240 candidate calls and 215 recovery calls, yielding Normalized Input Cost 1.858. This is consistent with the fixed-state candidate-generation decomposition rather than evidence that selective routing is intrinsically more expensive.

The candidate-free fixed-state analysis does not establish what its on-policy trajectories, costs, or final task outcomes would be. Moreover, unconditional and threshold-based policies were collected in separate rollout batches, and routing changes the distribution of later states. These small-cohort on-policy results should therefore be interpreted as deployment diagnostics, not as an extension of the fixed-state CRC guarantee to policy-induced state distributions.

G Ablation and Portability Details

G.1 Calibration-Unit Ablation Details

Section 4.4 separates two consequences of flattening dependent decisions into nominal calibration examples. Let n denote the number of calibration units and m the number of controlled decisions. *Formal* uses unit-equal loss and applies the CRC correction using n. *Diagnostic A* preserves the same unit-equal loss but substitutes m for n throughout the corrected criterion. *Naive* additionally replaces the unit-equal loss by a decision-equal average. Thus, Formal–Diagnostic A isolates the calibration-count effect, whereas Diagnostic A–Naive isolates the effect of loss weighting. The latter two are diagnostic constructions rather than certified procedures for dependent decisions.

Count and weighting effects. At $n_{\mathrm{cal}} = 20$, the additive finite-sample term is

$$\frac{1}{n+1} = \frac{1}{21} \approx .0476.$$

Substituting decision counts for unit counts reduces this term to approximately $.004$–$.007$ in the displayed multi-decision configurations. On Retail, Formal–Diagnostic A changes FS and coverage by $-.059$ and $-.127$, whereas Diagnostic A–Naive changes both by only $+.002$. PersonaMem-v2 shows no weighting effect at the reported precision. On Airline, the count effect also remains larger, although weighting contributes $-.016$ FS and $-.022$ coverage.

In the displayed configurations, the count effect is larger than the weighting effect. This does not imply the same ordering for every workload.

Replication audit. Uniformly duplicating decisions within their original units adds no independent calibration information. Accordingly, the Formal procedure is invariant to such replication. In the frozen audit with $n_{\mathrm{cal}} = 25$ and $\alpha = .10$, Formal test FS remains $.0603$ under $1\times$, $2\times$, $5\times$, and $10\times$ replication. By contrast, duplicating each decision twice changes the decision-count diagnostic from $.0603$ to $.1060$. Thus, nominal decision count can alter the selected operating point even when the number of independent calibration units is unchanged.

Calibration size. The CRC correction weakens as the number of independent calibration units increases: the additive term decreases from $1/11 \approx .0909$ at $n = 10$ to $1/401 \approx .0025$ at $n = 400$. On expanded PrefEval with Llama at $\alpha = .10$, the resulting ReCoverMem and Empirical-risk operating points are nearly identical: FS/Cov. are $.091/.201$ and $.091/.203$, respectively.

A smaller correction does not imply monotonic changes in threshold, coverage, or empirical test FS because the empirical calibration loss also varies across sampled calibration units.

Structural equality check. Each PrefEval source contributes one controlled decision. Hence $m = n$, and source-equal and decision-equal losses coincide on the unreplicated data. PrefEval therefore cannot distinguish Formal, Diagnostic A, and Naive through count or weighting changes alone; the multi-decision workloads provide the corresponding separation.

Explicit diagnostic criteria. Let $\widehat{L}_U$ be the mean of within-unit losses and $\widehat{L}_D$ the mean over all m decisions. The three criteria are

$$\text{Formal}: \quad \frac{n\widehat{L}_U(\tau)+1}{n+1} \le \alpha,$$

$$\text{Diagnostic A}: \quad \frac{m\widehat{L}_U(\tau)+1}{m+1} \le \alpha,$$

$$\text{Naive}: \quad \frac{m\widehat{L}_D(\tau)+1}{m+1} \le \alpha.$$

All use strict thresholding and the same unit-equal test metrics. Thus, the count substitution changes both the multiplier of empirical loss and the additive term; it is not merely subtraction of two additive penalties.

Decomposition population and draws. The main decomposition uses Llama-3.1-8B-Instruct at $\alpha = .10$ and 20 calibration units. Retail and Airline average 200 paired calibration draws; PersonaMem-v2 uses its single full-pool subset. The same frozen test population is used within each workload. The displayed contrasts are rounded point estimates without uncertainty intervals.

Calibration-size diagnostics. Figure A1 compares Formal, Diagnostic A, and Naive across calibration sizes at $\alpha = .10$. Formal uses unit-equal calibration loss and the unit count n; Diagnostic A substitutes the decision count m while preserving the loss weighting; Naive additionally uses decision-equal calibration loss. The joint FS and coverage results show how these choices change the selected operating point. The PrefEval results come from two earlier sampling populations and are separate from the current 400-source calibration experiment.

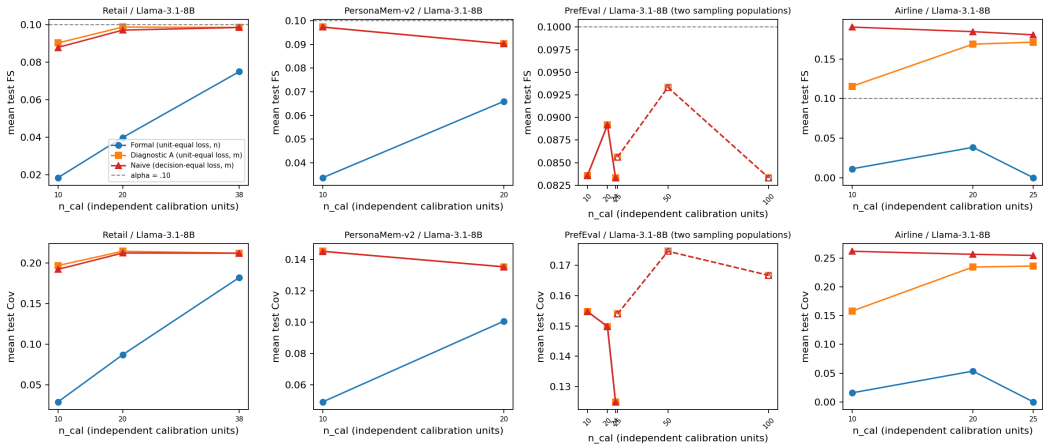


Figure A1: **Risk and coverage across calibration sizes.** The top row reports mean unit-equal empirical test FS; the bottom row reports mean unit-equal trust coverage. Formal, Diagnostic A, and Naive separate the effects of calibration counts and loss weighting. The dashed horizontal line in the FS panels marks $\alpha = .10$. Increasing calibration size need not change the empirical operating point monotonically. The PrefEval panel contains two distinct sampling populations from the earlier subset study; its solid and dashed curves must not be interpreted as a single calibration-size trajectory.

G.2 Host-Memory Portability Protocol

The complete results appear in Table 3 in Section 4.4. All four hosts use the same Retail task split and downstream Llama-3.1-8B-Instruct backbone. Each host produces its own memory representation, retrieved evidence, and memory-route labels, followed by separate scorer fitting and threshold calibration. Scorer fitting uses only predictor-training units, and calibration uses no final-test outcomes.

Three-Layer Memory uses the frozen bounded v2 implementation. The reported A-Mem comparison follows the strict convention $s_t > \tau$, with ties recovering, including threshold selection, final-test routing, and the corresponding exceedance diagnostic. The audit recomputes these quantities from frozen scores without repeating memory construction or scorer fitting.

The host comparison measures portability of the procedure with host-specific fitting. It does not transfer one frozen scorer or numerical threshold across hosts. Because hosts can induce different controlled-decision populations, shared task splits do not establish a causal effect of changing the memory implementation alone.

G.3 Retail Candidate-Generation Decomposition

Table A15 expands the main-text A/B/C table with baseline costs and absolute cohort token totals. A is the canonical candidate-based policy. B retains A's routes and removes candidate charges on Recover decisions, so it is an accounting counterfactual. C fits and calibrates a candidate-free scorer and generates only the selected answer branch.

A–B removes 1,947,301 input tokens without changing selected outcomes. Its normalized reduction is .7869 (paired episode-bootstrap 95% CI $[.7402, .8265]$). C has normalized input cost .9953, giving a point-estimate saving of approximately 0.47% relative to Always Recover. FS and coverage also change from $.075/.182$ to $.060/.152$; equality of rounded routed success ($.356$) is not a claim that the controllers make identical decisions. Absolute token totals are cohort sums, whereas normalized input cost is a mean of per-episode ratios. Their numerical savings need not coincide.

History length and context pressure. In the recorded Retail analysis, coverage declines across history-length groups, and all 439 evaluated histories fit the available context window; maximum occupancy is 17.6%. This is an observational analysis of the configured routes, not an intervention on compression pressure. Realized savings depend jointly on relative branch costs and which decisions are selected for Trust. Neither this analysis nor the fixed-state candidate-free decomposition establishes corresponding on-policy costs or final task outcomes.

Table A15: **Candidate-generation cost decomposition on fixed-state Retail.** All conditions use Llama-3.1-8B-Instruct on the same 40 final-test episodes and 439 controlled decisions. Calibrated policies use $\alpha = .10$ and strict routing $s_t > \tau$. FS, Cov., and routed success are episode-equal. Normalized Input Cost averages per-episode input-token ratios against Always Recover; absolute input tokens are summed over the cohort. Costs are computed offline from frozen branch-token logs. Output tokens, host-memory maintenance, and training, calibration, and evaluation-only calls are excluded. B preserves A's routes while removing candidate charges on Recover decisions; it is an accounting counterfactual. C and Random + CRC charge only the selected answer-generation branch.

Condition	FS	Cov.	Routed success	Normalized Input Cost	Routing input tokens
A. Candidate-based	.075	.182	.356	1.7806	4,358,384
B. Selected-branch accounting	.075	.182	.356	.9937	2,411,083
C. Candidate-free	.060	.152	.356	.9953	2,414,678
Random + CRC (selected branch)	.072	.115	.362	.9947	2,412,486
Always Recover	.000	.000	.362	1.0000	2,425,636
Always Trust	.618	1.000	.382	.9492	2,299,861

H Discussion and Limitations

Risk control is distinct from downstream utility. ReCoverMem controls the marginal risk of trusting an operationally insufficient memory route; it does not guarantee that recovery repairs the decision or improves final task success. Operational sufficiency is defined relative to the frozen backbone, evidence, decoding protocol, and evaluator, rather than attributing failure to compression alone. Our paired and on-policy evaluations therefore treat memory sufficiency, recovery effectiveness, and end-task utility as distinct quantities.

Validity depends on the calibration population. The CRC guarantee assumes exchangeability of calibration and future units after fixing the scorer and decision-generation protocol, and is marginal over the calibration sample and a future unit. It does not control error conditional on TRUST, require every finite test set to satisfy the target, or automatically extend to routing-induced on-policy state distributions. Unit-level aggregation handles within-unit dependence but does not establish exchangeability across units or under distribution shift.

Benefits depend on the operating regime. Memory-sufficiency prevalence constrains the available risk–coverage region, while scorer quality and calibration determine the selected operating point. Efficiency additionally depends on which decisions are trusted, the relative costs of TRUST and RECOVER, and candidate generation overhead. NarrativeXL shows measurable token savings with a success trade-off, whereas candidate-free Retail is nearly cost-neutral. Thus, the practical benefit of risk-controlled memory trust depends on both the recovery mechanism and the deployment regime.