AGENT MEMORY EVICTION HAS A BLAST RADIUS: DISTRIBUTION-FREE TAIL-RISK CERTIFICATES

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language model (LLM) agents rely on persistent memory for long-term tasks, and limited storage makes memory eviction inevitable. One eviction action can remove evidence needed by many future queries, a magnitude that harmful-decision frequency does not record. We define the *blast radius* of an eviction action as the number of future queries that lose required evidence because of it, and introduce a distribution-free framework that treats each conversation as one calibration unit, allowing arbitrary dependence among its actions. Across independent and identically distributed (i.i.d.) conversations, the framework certifies an eviction threshold for catastrophe probability (CAT), the probability that some eviction harms at least a chosen number of queries, jointly with the harmed-query rate, and remains valid when attribution drift makes risk non-monotone in the threshold. We prove that, even when no eviction harms any query, no valid conversation-level bound certifies mixture conditional value-at-risk (MIX-CVaR) at tail level η from fewer than about $1/(1-\eta)$ times the conversations CAT needs. On LongMemEval, certifying only the harmed-query rate at .25 leaves the concentration of harm uncertified, and 24.3% of test streams contain an eviction that harms multiple queries; certifying CAT at .20 jointly guarantees this share, which is 8.5% on test streams, at the cost of deleting 15.8% instead of 38.3% of memory. On memories extracted by Mem0, the joint rule certifies deleting 27.8% of memories with 8.0% of held-out streams affected. At the thresholds certified by CAT ($z^* = 2$) at .10 and .20, question-answering accuracy changes by about one point on average; above them, answers to queries that lost evidence turn from correct to wrong 31 times for every 2 reverse changes.

1 INTRODUCTION

As a large language model (LLM) agent interacts with a user over time, it accumulates constraints, preferences, and prior decisions that may matter later. Persistent memory keeps this evolving context available across sessions (Packer et al., 2023; Chhikara et al., 2025), but a bounded memory store must eventually discard some of it. We study threshold-based compaction: when compaction is triggered, the agent scores eligible memory entries and removes all entries below a retention threshold. Because future queries can depend on overlapping parts of memory, one compaction can destroy evidence needed by several of them. We call the number of future queries whose required evidence one action destroys within a chosen horizon its **blast radius**.

Existing distribution-free methods control prescribed mean and tail risks at user-chosen levels (Bates et al., 2021; Angelopoulos et al., 2024; Chen et al., 2025). Yet applying them directly to eviction decisions does not match this batched, history-dependent process. First, harmful-decision frequency reduces every damaging compaction to a binary event, so an action that harms one future query is indistinguishable from one that harms many (Figure 1). Blast radius restores this missing magnitude. Further, raising the threshold can delete evidence earlier and redistribute downstream harm across actions, causing measured action-level tail severity to fall even when query harm does not decrease. We call this *attribution drift*. Finally, compactions within a conversation share history and future evidence requirements, so their losses cannot be treated as independent calibration samples. Consequently, frequency-based control, monotone threshold search, and decision-level calibration do not guarantee valid certification of the tail risk of memory eviction.

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

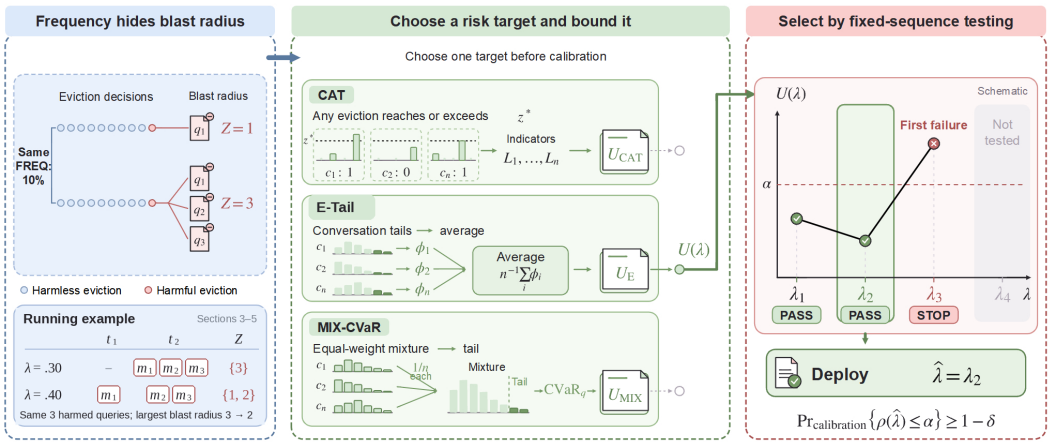

Figure 1: From eviction harm to a certified policy. Left: harmful-decision frequency hides differences in blast radius; the lower box is the running example of Sections 3–5. Center: choose one risk target before calibration and bound it using independent conversations. Right: test thresholds in a fixed order, stopping at the first failure; deploy the last passing threshold or abstain. The threshold curve is schematic.

To address these failures, our framework follows the three stages in Figure 1. First, it fixes the operational target. Catastrophe probability (CAT) asks whether any eviction in a conversation reaches a chosen blast radius and is certified jointly with the harmed-query rate; expected within-conversation tail (E-Tail) and mixture conditional value-at-risk (MIX-CVaR) average severe tails within conversations or over their equal-conversation mixture, and we characterize how many conversations each needs. Second, each independent, deployment-representative conversation contributes one bounded loss or empirical distribution, yielding a high-confidence upper bound $U(\lambda)$ for every candidate threshold. Finally, fixed-sequence testing (Angelopoulos et al., 2025) examines candidates in a prespecified order and stops at the first bound above tolerance α, deploying the last passing threshold or abstaining. Per-threshold bounds and the fixed order ensure validity without assuming monotone risk. This design separates the operational target, the conversation-level calibration unit, and deployable-threshold selection. This framework provides long-term agent memory with a principled way to measure, control, and certify the downstream risk of eviction. Our contributions are:

1. **Problem formalization.** We define action-level blast radius through delayed query-level evidence loss, and a window variant that splitting compactions cannot lower (Section 3).
2. **Structural-obstacle diagnosis.** We establish frequency–severity separation, attribution drift, the effect of splitting compactions, and tight relations among five risk measures, and we motivate conversation-level calibration (Sections 4 and 5).
3. **Certification framework.** We certify CAT jointly with the harmed-query rate, with threshold selection that remains valid for non-monotone risk and under anytime stopping, and prove a zero-event floor showing that, even without observed harm, MIX-CVaR needs a factor $1/(1-\eta)$ more conversations than CAT (Section 5).
4. **Empirical validation.** We test validity on synthetic and benchmark populations, show that frequency control leaves most queries without evidence on independent streams with distractors, certify eviction on memories extracted by Mem0, and measure question answering (QA) at every threshold our rules certify in the main setting (Section 6).

2 Related Work

Agent memory. Persistent memory uses hierarchies (Packer et al., 2023; Kang et al., 2025), extraction and consolidation (Chhikara et al., 2025), and graphs (Rasmussen et al., 2025; Xu et al., 2025). Outcome-based forgetting (Simsek, 2026) and rate-distortion analysis (Colaco & Lahjouji, 2026) study constrained retention. LoCoMo and LongMemEval (Maharana et al., 2024; Wu et al.,

2025) evaluate long-term evidence use; KV-cache eviction (Zhang et al., 2023; Xiao et al., 2024; Li et al., 2024) concerns transient inference state. We certify the tail severity of downstream query harm from destructive memory eviction.

Distribution-free risk control. Conformal prediction and high-probability risk bounds underpin distribution-free calibration (Vovk et al., 2005; Angelopoulos & Bates, 2023; Bates et al., 2021). Conformal Risk Control (CRC) controls the mean of a monotone loss, Learn-then-Test selects candidates by multiple testing without monotonicity, and Quantile Risk Control bounds loss quantiles rather than means (Angelopoulos et al., 2024; 2025; Snell et al., 2023). Hierarchical and non-exchangeable methods (Dunn et al., 2023; Barber et al., 2023), non-monotone mean-risk control (Angelopoulos, 2026), and trajectory-level agent calibration (Opoku & Banahene, 2026) address related structure. Here future queries determine earlier actions' severity, and thresholds change its attribution. We construct risks and bounds at the conversation level; Learn-then-Test supplies policy selection.

Tail risks. VaR specifies a quantile; CVaR averages the worst fraction of losses (Rockafellar & Uryasev, 2000). Quantile, prompt, and distortion-risk control (Snell et al., 2023; Zollo et al., 2024; Chen et al., 2025) characterize high-loss outcomes beyond the mean, while online CVaR control (Chen et al., 2026) gives a sequential perspective. We distinguish catastrophe probability, expected within-conversation tail, and conversation-mixture tail; each has a different guarantee and calibration cost.

3 Problem Setup: Blast Radius of Memory Eviction

3.1 Memory Stream and Eviction Policy

An agent maintains memory entries while processing conversation c. At an eviction decision time τ, $\mathcal{B}_\tau$ denotes the entries considered for deletion. The policy assigns each $m \in \mathcal{B}_\tau$ a retention score $s_\tau(m)$, with larger values favoring retention. Threshold λ deletes

$$E_\tau(\lambda) \;=\; \{\, m \in \mathcal{B}_\tau : s_\tau(m) < \lambda \,\}. \tag{1}$$

Entries removed at τ form one eviction action and do not return to memory. Our structural results assume that $s_\tau(m)$ depends on the stream, the ingestion time of m, and τ, but not on earlier evictions (Assumption 1), and that decision times and eligibility do not depend on λ (Assumption 2); the certificates need neither (Appendix D). For a fixed pool $\mathcal{B}_\tau$ and scores, increasing λ expands $E_\tau(\lambda)$; across time, it can also move a deletion earlier, shrink later pools, and redistribute harm across actions, so tail risk need not be monotone in λ (Section 4).

3.2 Query Harm and Blast Radius

For each query q arriving at time t_q, let $\mathcal{S}_q$ denote its required-evidence set, and let $\mathrm{Ev}_{<t_q}(\lambda) = \bigcup_{u<t_q} E_u(\lambda)$ denote the entries evicted before its arrival. The query-level evidence-loss indicator is

$$\ell_q(\lambda) = \mathbf{1}\big[\mathcal{S}_q \cap \mathrm{Ev}_{<t_q}(\lambda) \neq \varnothing\big]\,. \tag{2}$$

Thus, $\ell_q(\lambda) = 1$ when eviction removes at least one entry required by the query. It measures evidence loss; downstream answer quality is evaluated separately. For a horizon of h turns, the **blast radius** of the eviction action at time τ is

$$Z_\tau^{(h)}(\lambda) = \sum_{q:\, \tau<t_q\le\tau+h} \mathbf{1}[\mathcal{S}_q \cap E_\tau(\lambda) \neq \varnothing]\,. \tag{3}$$

It counts the distinct future queries that require at least one entry removed by this action. A query counts once for a given action, even if the action removes several entries it requires. If different actions remove different required entries for the same query, that query contributes once to each action; consequently, $\sum_\tau Z_\tau$ can exceed the number of queries with evidence loss. Unless stated otherwise, the horizon extends to the end of the conversation, and we write $Z_\tau(\lambda)$. Increasing h can only add future queries, so $Z_\tau^{(h)}(\lambda)$ is nondecreasing in h. An evaluator may also fix windows of W turns and count, for each window, the distinct future queries that lose required evidence to any action inside it. This *window blast radius* equals Z_τ when a window holds one action and cannot be lowered by splitting a compaction (Appendix A.3).

Running example. A conversation stores memories m_1, m_2, m_3, and three later queries q_1, q_2, q_3 each require one of them. Compaction runs at $t_1 < t_2$, before the queries. At $\lambda = .30$ the action at t_2 deletes all three memories, so $Z_{t_2} = 3$ and every query loses evidence (Figure 1, lower left).

3.3 Risk Targets

We measure eviction harm at three levels: how often an eviction action is harmful, how many queries lose required evidence, and how much harm is concentrated in a single action. Let $\mathcal{D}_c(\lambda) = \{\tau : E_\tau(\lambda) \neq \varnothing\}$ denote the evicting actions in conversation c, and let $\mathcal{Q}_c$ denote its queries. The harmful-decision frequency and harmed-query rate are

$$f_c(\lambda) = \frac{\sum_{\tau \in \mathcal{D}_c(\lambda)} \mathbf{1}[Z_\tau(\lambda) > 0]}{|\mathcal{D}_c(\lambda)|}, \qquad r_c(\lambda) = \frac{\sum_{q \in \mathcal{Q}_c} \ell_q(\lambda)}{|\mathcal{Q}_c|}, \tag{4}$$

Both are defined as zero when their denominator is zero. The first measures the fraction of evicting actions that harm at least one future query; the second measures the fraction of queries that lose required evidence. Their population-level counterparts are $\rho_{\mathrm{F}}(\lambda) = \mathbb{E}_c[f_c(\lambda)]$ and $\rho_{\mathrm{H}}(\lambda) = \mathbb{E}_c[r_c(\lambda)]$, with each conversation receiving equal weight.

These rates describe harm from action- and query-level perspectives, but neither preserves how affected queries are concentrated under individual eviction actions. Severity targets therefore operate on the action-level blast radii $\{Z_\tau(\lambda) : \tau \in \mathcal{D}_c(\lambda)\}$. CAT uses raw counts and controls $\Pr_c(\max_\tau Z_\tau(\lambda) \geq z^*)$, where the maximum is zero when no action evicts. E-Tail and MIX-CVaR use capped severities

$$\tilde{Z}_\tau(\lambda) = \frac{\min\{Z_\tau(\lambda), B\}}{B} \in [0, 1],$$

where values at or above B are represented by one. Section 5 turns these conversation-level targets into certificates for policy selection.

4 Why Standard Risk Control Fails

4.1 Targets Constrain One Another Only Loosely

Harmful-decision frequency retains only whether $Z_\tau > 0$, and the harmed-query rate retains only how many queries lose evidence. E-Tail averages each conversation's CVaR of capped severities, and MIX-CVaR takes the CVaR of their equal-conversation mixture (Section 5.2).

Proposition 1 (Relations among targets)**.** *Use the full-conversation horizon, let every conversation have at most K evicting actions and Q queries, let ρ_{F}, ρ_{H}, ρ_{C}, ρ_{E}, and ρ_{M} be the harmful-decision frequency, the harmed-query rate, CAT with an integer $z^* \leq B$, E-Tail, and MIX-CVaR at a common tail level η, and let $\tilde{z} = z^*/B$. Then*

$$\rho_{\mathrm{C}} \leq \min\Big\{K\rho_{\mathrm{F}},\ \tfrac{Q}{z^*}\rho_{\mathrm{H}}\Big\},\ \ \rho_{\mathrm{H}} \leq K\rho_{\mathrm{F}},\ \ \rho_{\mathrm{F}} \leq Q\rho_{\mathrm{H}},\ \ \tfrac{1}{B}\min\Big\{1, \tfrac{\rho_{\mathrm{F}}}{1-\eta}\Big\} \leq \rho_{\mathrm{M}} \leq \min\Big\{1, \tfrac{\rho_{\mathrm{F}}}{1-\eta}\Big\},$$

$$\tilde{z}\min\Big\{1, \tfrac{1}{(1-\eta)K}\Big\}\rho_{\mathrm{C}} \ \leq\ \rho_{\mathrm{E}} \ \leq\ \min\Big\{\rho_{\mathrm{M}},\ \rho_{\mathrm{C}} + (1-\rho_{\mathrm{C}})\big(\tilde{z} - \tfrac{1}{B}\big)\Big\}.$$

Each inequality holds with equality for some distribution of conversations that satisfies these constraints, and without a bound on raw blast radius, equal frequencies allow the CVaR of raw blast radius to differ by any amount.

Apart from $\rho_{\mathrm{E}} \leq \rho_{\mathrm{M}}$, the factors grow with K, Q, and B, so a bound on one target leaves the others loose: on LongMemEval ($K = 12$, $Q = 10$), a harmed-query bound of .25 implies only CAT ($z^* = 2$) ≤ 1.25 (Appendix A.1).

4.2 Attribution Drift Breaks Monotone Calibration

Raising λ can delete a required memory earlier, changing how the resulting query losses are distributed across eviction actions. The same harmed queries may therefore produce a different collection of action-level blast radii. We call this *attribution drift*.

Proposition 2 (Attribution drift)**.** *Under Assumptions 1–2, let $\lambda_1 < \lambda_2$. (i) The maximum action-level blast radius can decrease from λ_1 to λ_2 while the same queries lose evidence. (ii) A query counted by action τ at λ_1 but not at λ_2 has every source removed at τ under λ_1 removed strictly earlier under λ_2; hence $Z_\tau(\lambda_2) \geq Z_\tau(\lambda_1) - \Delta_\tau$, where Δ_τ counts such drifted queries. (iii) If each entry's score does not decrease across its eligible decisions, as under one-shot review, every Z_τ, its maximum, and CAT are nondecreasing in λ, whereas E-Tail and MIX-CVaR can still fall when new low-severity actions dilute the tail.*

Running example. Let the scores at t_1 be $(.35, .50, .50)$ and all scores at t_2 be $.25$. At $\lambda = .40$, t_1 already deletes m_1 and t_2 deletes m_2, m_3. Every memory is deleted no later than at $\lambda = .30$, and the same three queries lose evidence, yet the severity sample changes from $\{3\}$ to $\{1, 2\}$: the maximum and the empirical CVaR_η for $\eta \in [1/2, 1)$ fall from 3 to 2.

Drift therefore stems from scores that decrease at a later eligible decision, as recency terms make them do; on LoCoMo and LongMemEval every decrease of a conversation's largest blast radius coincides with drift, and one-shot review removes them (Appendices A.2 and E.3).

5 Conversation-Level Tail Certification

5.1 Framework

Eviction actions within one conversation are dependent, and pooling all actions would give more weight to conversations with more evictions. We therefore use **one conversation as the independent calibration unit** (Appendix E.5). Calibration conversations are independent and identically distributed (i.i.d.) draws from the deployment distribution; actions within each may depend arbitrarily. Depending on the target, each conversation contributes one bounded summary or one empirical severity distribution (Appendix B).

Before examining calibration data, fix the policy family and finite threshold grid Λ, choose a risk target with tolerance α and failure probability δ, and specify the bound and test order. Evaluate the same n calibration conversations at every threshold. For true deployment risk $\rho(\lambda)$, construct a data-dependent upper bound $U(\lambda)$ satisfying

$$\Pr_{c_{1:n}}\{\rho(\lambda) \leq U(\lambda)\} \geq 1 - \delta \qquad \text{for each fixed } \lambda. \tag{5}$$

A threshold passes when $U(\lambda) \leq \alpha$; fixed-sequence testing makes selection valid (Section 5.4).

5.2 CAT and Joint Certification

CAT certifies the probability that a conversation contains an eviction whose blast radius reaches z^*; each conversation contributes one Bernoulli loss. For a chosen catastrophe size $z^* > 0$, define

$$L_c(\lambda) = \mathbf{1}\Big[\max_\tau Z_\tau(\lambda; c) \geq z^*\Big], \qquad \rho_{\mathrm{C}}(\lambda) = \mathbb{E}_c[L_c(\lambda)]. \tag{6}$$

The exact Clopper–Pearson limit from the losses $L_i = L_{c_i}$ bounds this mean; with no observed event it falls below α once $n \geq \log\delta / \log(1-\alpha)$, which is 22 conversations at $\alpha = \delta = .1$. CAT bounds how concentrated harm is, not how many queries lose evidence (Proposition 1), so our default rule certifies CAT jointly with the harmed-query rate ρ_{H}, whose per-conversation value $r_c \in [0, 1]$ the betting method of Waudby-Smith & Ramdas (2024) (WSR) bounds; a threshold passes only when both bounds meet their tolerances (Section 5.4).

Running example. With $z^* = 3$, the conversation contributes the CAT loss $L_c = 1$ at $\lambda = .30$ and $L_c = 0$ at $\lambda = .40$; its harmed-query rate is 1 at both. Each target receives one bounded value per conversation, however its actions depend on one another, and the CAT curve may fall as λ grows (Section 5.4).

5.3 Mean-Type Tail Targets and Their Cost

E-Tail and MIX-CVaR average severe tails instead of thresholding them. For a tail level $\eta \in (0, 1)$, E-Tail averages each conversation's exact empirical CVaR of capped severities,

$$\phi_c(\lambda) = \widehat{\mathrm{CVaR}}_\eta\Big(\{\tilde{Z}_\tau(\lambda) : \tau \in \mathcal{D}_c(\lambda)\}\Big) \in [0, 1], \qquad \rho_{\mathrm{E}}(\lambda) = \mathbb{E}_c[\phi_c(\lambda)], \tag{7}$$

and WSR bounds its mean. The exact empirical CVaR averages the largest $(1-\eta)k$ of k capped severities, giving the boundary value fractional weight, and is zero when $k = 0$. MIX-CVaR takes the CVaR of the equal-conversation mixture of capped severities: with $\hat{F}_\lambda^{(c)}$ the empirical CDF within conversation c, or a point mass at zero if $\mathcal{D}_c(\lambda) = \varnothing$,

$$F_\lambda = \mathbb{E}_c[\hat{F}_\lambda^{(c)}], \qquad \rho_{\mathrm{M}}(\lambda) = \mathrm{CVaR}_\eta(F_\lambda). \tag{8}$$

The Rockafellar–Uryasev representation (Rockafellar & Uryasev, 2000; 2002) reduces ρ_{M} to bounded conversation-level means on a finite grid Γ, each bounded by WSR at level $\delta/|\Gamma|$ (Appendix B.3).

Theorem 1 (Conversation-level certificates)**.** *For fixed λ and i.i.d. calibration conversations, the Clopper–Pearson bound from $L_i(\lambda)$, the WSR bound from $\phi_{c_i}(\lambda)$, and the Bonferroni-corrected dual over Γ (Appendix B.3) are $(1-\delta)$ upper confidence bounds for ρ_{C}, ρ_{E}, and ρ_{M}, without independence among actions within a conversation (Appendix B).*

Proposition 3 (Zero-event floor)**.** *Let $z^* \le B$, and let U depend on n i.i.d. conversations only through each conversation's empirical distribution of capped severities, as all three certificates do, and satisfy $\Pr\{\rho \le U\} \ge 1-\delta$ under every distribution of conversations. On a sample in which no action harms any query, $U \ge 1-\delta^{1/n}$ for CAT and E-Tail, and $U \ge \min\{1, (1-\delta^{1/n})/(1-\eta)\}$ for MIX-CVaR.*

Even when no action harms any query, certifying MIX-CVaR at α therefore needs $n \ge \log\delta/\log(1-(1-\eta)\alpha) \approx \log(1/\delta)/((1-\eta)\alpha)$ conversations, a factor of about $1/(1-\eta)$ more than CAT (Appendix C.1). At $n = 30$ and $\delta = .1$, no such bound on MIX-CVaR at $\eta = .9$ falls below .739 on a harm-free sample, so on our benchmark CAT is the certificate the data can support, while MIX-CVaR passes on synthetic pools from 200 conversations (Section 6.2).

5.4 Non-Monotone Threshold Selection

Order the threshold grid in advance as $\lambda_{(1)} < \lambda_{(2)} < \cdots$ and apply the fixed-sequence procedure of Learn-then-Test (Angelopoulos et al., 2025). Test each threshold by comparing $U(\lambda_{(j)})$ with α. After a pass, move to the next threshold; **at the first failure, stop** and return the last passing threshold. If none passes, abstain ($\hat{\lambda} = \perp$).

Proposition 4 (Validity without monotonicity; after Angelopoulos et al., 2025)**.** *Suppose the test order is fixed before the calibration data are observed, and each per-threshold bound satisfies Eq. (5). Treat abstention as a no-eviction policy with $\rho(\perp) = 0$. Then, for any possibly non-monotone risk curve, the procedure's output $\hat{\lambda}$ satisfies*

$$\Pr\Big\{\rho(\hat{\lambda}) \le \alpha\Big\} \ge 1-\delta.$$

Proof: selecting an unsafe threshold requires passing the first unsafe threshold in the fixed order, which occurs with probability at most δ by Eq. (5) (Appendix B.4). Several targets can share one threshold without splitting δ: a candidate passes only when every bound meets its tolerance. Abstention means that no threshold is certified at this tolerance and sample size; a system that must reclaim storage then relaxes α or annotates more conversations, which the anytime-valid version supports.

With time-uniform betting confidence sequences (Waudby-Smith & Ramdas, 2024), the same selection stays valid when conversations are annotated one at a time and calibration stops at any data-dependent time (Proposition 6, Appendices B.6 and E.6).

6 Experiments

Setup. We use 10 LoCoMo conversations (1,531 usable queries) and build 50 session-disjoint streams from LongMemEval-S (479 usable queries). Questions that share an answer session stay in one stream, each stream keeps only the answer sessions of its questions, and each question is asked at its benchmark question date. Streams share no session and serve as independent calibration units. A second LongMemEval population keeps every distractor session in the first stream that uses it, so its 50 streams are about 18 times longer. A third runs Mem0 (Chhikara et al., 2025) on the main

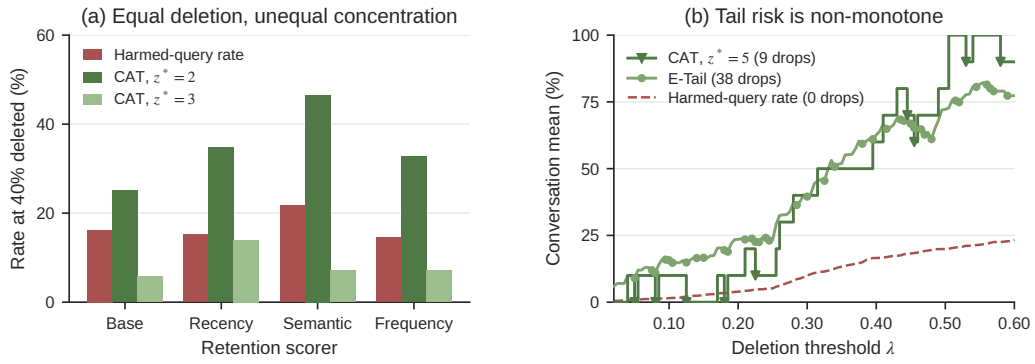

Figure 2: Mean harm and concentration. (a) LongMemEval, four retention scorers at 40% of memory deleted (50-stream rates, interpolated in deletion). (b) LoCoMo conversation-level CAT ($z^* = 5$) and E-Tail over 117 thresholds, with the harmed-query rate for reference; markers show decreases as the threshold increases.

streams and applies the eviction policy to the 10,636 memories it extracts (Appendix E.10). Each of 20 LongMemEval splits uses 30 streams for calibration and 20 for testing. All certificates use failure probability $\delta = .1$ (Appendix G).

6.1 Do Frequency and Mean Harm Control Concentration?

Finding. *No. Frequency control can leave most queries without evidence, and equal harm can hide very different concentration.*

• On streams with distractors, frequency at .15 allows 5.7% harmful actions while 68.1% of queries lose evidence.

On these streams, CAT ($z^* = 2$) and the harmed-query rate stop below a score cluster at .25 and, with an embedding scorer, abstain (Appendices E.8 and E.9). At 40% of LongMemEval memory deleted, three scorers harm 14.5–16.2% of queries, yet 5.8–13.9% of streams contain an action harming three or more (Figure 2a). Averages over queries or actions do not retain how many queries a single action harms, as Proposition 1 quantifies. On LoCoMo, CAT and E-Tail fall at some threshold increases while the harmed-query rate never does (Figure 2b): an earlier deletion redistributes the same losses across actions, as in Proposition 2, so threshold selection cannot assume monotone risk.

6.2 Are the Certificates Safe?

Finding. *Yes. Unsafe selections stay below the 10% budget, including on a benchmark population with non-monotone risk.*

• Unsafe-selection rates are 0–2.0% on synthetic pools (Figure 3a) and at most 2.6% on the LongMemEval population.

More conversations buy power, not safety: CAT returns a threshold in every repetition from 200 conversations, as does MIX-CVaR (Appendix E.4).

6.3 What Does the Certified Target Change?

Finding. *Which risk is guaranteed. Each certificate bounds only the risk it names, so CAT and the harmed-query rate are certified together.*

• With the harmed-query rate alone at .25, 24.3% of test streams contain an action harming two or more queries; the joint rule with CAT at .20 guarantees this share and holds it at 8.5%, deleting 15.8% instead of 38.3% (Table 1).

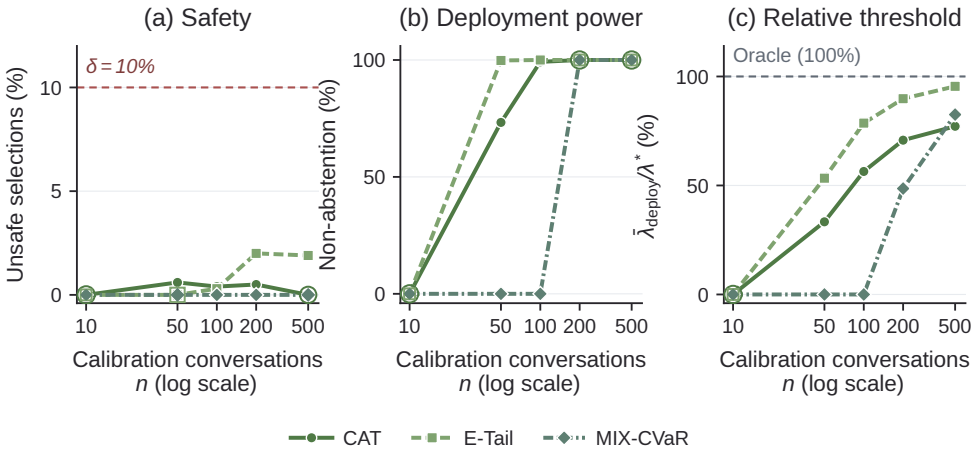

Figure 3: Safety and power across 1,000 i.i.d. repetitions: unsafe selection, non-abstention, and deployed threshold relative to each target's largest safe grid threshold (abstention is zero). Configurations are in Appendix E.4.

Table 1: Certified selections on LongMemEval ($\delta = .1$), means over 20 calibration/test splits. Test columns give the share of queries that lose evidence and the share of streams with an action harming at least z^* queries. Bold marks each row's certified risk where it is a column; MIX-CVaR uses $\eta = .9$ and $B = 10$. Appendix E.1 lists further targets and tolerances.

Certified target	α	Deleted	Harmed queries	CAT, z^*=2	CAT, z^*=3
CAT (z^*=2)	.10	8.1%	2.8%	**2.8%**	0.8%
CAT (z^*=2)	.20	15.8%	5.1%	**8.5%**	3.0%
CAT (z^*=2) + harmed queries	.20 + .25	15.8%	**5.1%**	**8.5%**	3.0%
Harmed-query rate	.15	16.5%	**5.4%**	7.8%	2.8%
Harmed-query rate	.25	38.3%	**15.9%**	24.3%	6.8%
MIX-CVaR	.30, .45	abstains on every split (floor .739)			

On Mem0 memories the pattern repeats: CAT at .20 deletes 27.8% with 8.0% of test streams affected, against 44.3% and 27.0% for the harmed-query rate alone at .25 (Appendix E.10).

Within the one-parameter family, every rule selects a point on one curve: the harmed-query rate at .15 and CAT at .20 delete 16.5% and 15.8% with similar realized risks, so the target fixes which risk is guaranteed, not a better operating point. A richer family lets the certificate reach a better point: over scorer × threshold, CAT at .30 deletes 29.0% of memory, as with the base scorer alone (29.2%), while the share of test streams with a multi-query eviction falls from 18.3% to 13.3% (Appendix E.2).

6.4 Do Certified Thresholds Preserve Answers?

Finding. *At the thresholds certified by CAT ($z^* = 2$), accuracy on held-out streams changes by about one point on average; above these thresholds, answers to queries that lost evidence turn wrong.*

- Test accuracy changes on average by -0.8 and -1.2 points under CAT ($z^* = 2$) at .10 and .20, with split-level changes from -4.2 to $+1.5$ (Figure 4b).

These thresholds delete 8.1% and 15.8% of memory, and across all rules the loss broadly grows with deletion (Table 12).

Each split scores its certified threshold on its test queries, with a reader and a separate judge model (Appendix F.1). Answers degrade through lost evidence: correct-to-wrong changes concentrate on

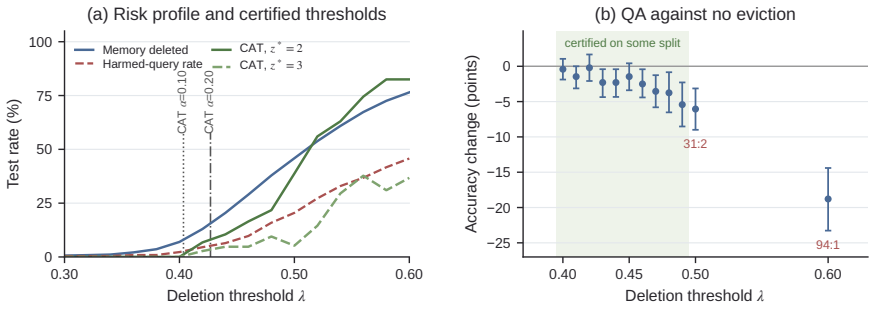

Figure 4: LongMemEval at and beyond certified thresholds. (a) Test rates over the 20 splits; vertical lines mark the mean thresholds certified by CAT ($z^* = 2$) at .10 and .20. (b) QA accuracy change against no eviction at every certified threshold (shaded) and beyond, with 95% stream-bootstrap intervals; labels give correct-to-wrong versus wrong-to-correct changes among queries that lost evidence.

queries whose evidence was deleted, while the other queries change in both directions at similar rates.

Choosing the tolerance. The tolerance trades deletion against concentration and answer quality: raising CAT ($z^* = 2$) from .10 to .20 raises deletion from 8.1% to 15.8% and the test CAT risk from 2.8% to 8.5%, while QA accuracy changes by -0.8 and -1.2 points; the harmed-query rate alone at .25 deletes 38.3% and costs 4.8 points (Table 12). The tolerance and z^* are fixed before calibration, for instance from requirements or a separate pilot split; with the anytime-valid version, the certified threshold then relaxes as more conversations are annotated.

7 Limitations

Our guarantees require evidence within a horizon chosen in advance and independent conversations representative of deployment; retrieval logs from an uncompressed shadow memory can replace annotation at some cost in deletion (Appendix E.7). On Mem0 we build the memory store once and then evict, so eviction does not feed back into later extraction. An embedding scorer reproduces the main-setting results, but with distractor backgrounds both scorer types concentrate scores near one threshold, so long streams need a recency scale that grows with the stream (Appendices E.8 and E.9). MIX-CVaR needs more conversations than our benchmarks provide (Proposition 3). History-dependent policies require replay at each threshold, and deployment shifts require recalibration.

8 Conclusion

Blast radius counts the future queries harmed by one eviction, a magnitude that frequency and mean harm do not retain. Certifying CAT jointly with the harmed-query rate, with one conversation per calibration unit, bounds both how concentrated and how widespread harm is, with dependent actions and non-monotone risk; even without observed harm, MIX-CVaR needs about a factor $1/(1-\eta)$ more conversations. At the thresholds certified by CAT ($z^* = 2$), QA accuracy on held-out streams changes by about one point on average, while answers degrade once evidence loss spreads.

Reproducibility Statement

The experiment archive includes source code and configurations, per-split calibration outputs, the Mem0 memories, and per-question answers and judgments. These records support regeneration of the reported summaries and figures. The trajectory-coupling assumptions and exact shared-sweep construction are given in Appendices A.4 and A.5.

Replay and calibration are deterministic conditional on fixed inputs, configurations, and random-number states. The reproducibility manifest records official dataset URLs, file sizes, MD5 and SHA-256 fingerprints, all query exclusions, and the exact lexical stopword set. Dataset adapters reconstruct every conversation stream from those author-released snapshots. For hosted reader and judge calls, the archived per-question outputs and parsers define the QA sample analyzed in Appendix F.

AI Use Statement

Generative AI tools were used for (i) writing assistance—language checking, copy-editing, and stylistic polishing of author-drafted text—and (ii) research support—literature search and screening, framing discussion, experimental-condition design, code review and debugging, and result-file inspection. No text, result, or claim was accepted without author verification: every number was recomputed from the released artifact and every quotation checked against its source. The reader and judge models used in the reported QA experiments are described in Appendix G.4.

References

A. N. Angelopoulos. Conformal risk control for non-monotonic losses. *arXiv:2602.20151*, 2026.

A. N. Angelopoulos and S. Bates. Conformal prediction: A gentle introduction. *Foundations and Trends in Machine Learning*, 16(4):494–591, 2023.

A. N. Angelopoulos, S. Bates, E. J. Candès, M. I. Jordan, and L. Lei. Learn then test: Calibrating predictive algorithms to achieve risk control. *Annals of Applied Statistics*, 19(2):1641–1662, 2025.

A. N. Angelopoulos, S. Bates, A. Fisch, L. Lei, and T. Schuster. Conformal risk control. In *ICLR*, 2024.

R. F. Barber, E. J. Candès, A. Ramdas, and R. J. Tibshirani. Conformal prediction beyond exchangeability. *Annals of Statistics*, 51(2):816–845, 2023.

S. Bates, A. N. Angelopoulos, L. Lei, J. Malik, and M. I. Jordan. Distribution-free, risk-controlling prediction sets. *Journal of the ACM*, 68(6):43:1–43:34, 2021.

C. Chen, J. Shen, Z. Deng, and L. Lei. Conformal tail risk control for large language model alignment. In *ICML*, PMLR 267:8955–8978, 2025.

C. Chen, J. Shen, Z. Deng, and L. Lei. Adversarially robust control of conditional value-at-risk via Rockafellar–Uryasev conformal inference. *arXiv:2606.00320*, 2026.

P. Chhikara, D. Khant, S. Aryan, T. Singh, and D. Yadav. Mem0: Building production-ready AI agents with scalable long-term memory. In *ECAI*, pages 2993–3000, 2025.

A. G. Colaco and N. Lahjouji. What to keep, what to forget: A rate-distortion view of memory compaction in LLMs and agents. *arXiv:2607.08032*, 2026.

R. Dunn, L. Wasserman, and A. Ramdas. Distribution-free prediction sets for two-layer hierarchical models. *Journal of the American Statistical Association*, 118(544):2491–2502, 2023.

J. Kang, M. Ji, Z. Zhao, and T. Bai. Memory OS of AI agent. In *EMNLP*, pages 25961–25970, 2025.

Y. Li, Y. Huang, B. Yang, B. Venkitesh, A. Locatelli, H. Ye, T. Cai, P. Lewis, and D. Chen. SnapKV: LLM knows what you are looking for before generation. In *NeurIPS*, 2024.

A. Maharana, D.-H. Lee, S. Tulyakov, M. Bansal, F. Barbieri, and Y. Fang. Evaluating very long-term conversational memory of LLM agents. In *ACL*, 2024.

J. Opoku and D. Banahene. ToolChain-CRC: Conformal risk control for agentic AI under retrieval and tool-use drift. *arXiv:2606.18467*, 2026.

C. Packer, S. Wooders, K. Lin, V. Fang, S. G. Patil, I. Stoica, and J. E. Gonzalez. MemGPT: Towards LLMs as operating systems. *arXiv:2310.08560*, 2023.

P. Rasmussen, P. Paliychuk, T. Beauvais, J. Ryan, and D. Chalef. Zep: A temporal knowledge graph architecture for agent memory. *arXiv:2501.13956*, 2025.

R. T. Rockafellar and S. Uryasev. Optimization of conditional value-at-risk. *Journal of Risk*, 2(3):21–41, 2000.

R. T. Rockafellar and S. Uryasev. Conditional value-at-risk for general loss distributions. *Journal of Banking & Finance*, 26(7):1443–1471, 2002.

B. Simsek. When to forget: A memory governance primitive. *arXiv:2604.12007*, 2026.

J. C. Snell, T. P. Zollo, Z. Deng, T. Pitassi, and R. Zemel. Quantile risk control: A flexible framework for bounding the probability of high-loss predictions. In *ICLR*, 2023.

V. Vovk, A. Gammerman, and G. Shafer. *Algorithmic Learning in a Random World*. Springer, 2005.

I. Waudby-Smith and A. Ramdas. Estimating means of bounded random variables by betting. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 86(1):1–27, 2024.

D. Wu, H. Wang, W. Yu, Y. Zhang, K.-W. Chang, and D. Yu. LongMemEval: Benchmarking chat assistants on long-term interactive memory. In *ICLR*, 2025.

G. Xiao, Y. Tian, B. Chen, S. Han, and M. Lewis. Efficient streaming language models with attention sinks. In *ICLR*, 2024.

W. Xu, Z. Liang, K. Mei, H. Gao, J. Tan, and Y. Zhang. A-MEM: Agentic memory for LLM agents. In *NeurIPS*, 2025.

Z. Zhang, Y. Sheng, T. Zhou, T. Chen, L. Zheng, R. Cai, Z. Song, Y. Tian, C. Ré, C. Barrett, Z. Wang, and B. Chen. H_2O: Heavy-hitter oracle for efficient generative inference of large language models. In *NeurIPS*, 2023.

T. P. Zollo, T. Morrill, Z. Deng, J. C. Snell, T. Pitassi, and R. Zemel. Prompt risk control: A rigorous framework for responsible deployment of large language models. In *ICLR*, 2024.

APPENDIX OVERVIEW

The appendices separate supporting material by purpose. Appendix A proves the structural claims and states the replay conditions. Appendix B gives the certificate constructions and statistical guarantees, and Appendix C studies their sample complexity. Appendix D audits replay assumptions and robustness. Appendix E provides extended blast-radius experiments, while Appendix F reports the end-task QA validation and robustness analyses. Appendix G collects the implementation settings and reproduction workflow. In the appendices, a *decision* with its severity Z_τ denotes the eviction action taken at decision time τ.

A PROOFS FOR THE STRUCTURAL CLAIMS

We use the definitions in Section 3.

A.1 RELATIONS AMONG TARGETS

Proof of Proposition 1. Fix a threshold and a conversation c with $k \le K$ evicting actions, $|\mathcal{Q}_c| \le Q$ queries, blast radii $Z_1, \dots, Z_k$, harmful-action share f_c, harmed-query share r_c, and $m_c = \max_i Z_i$ (zero if $k = 0$). Let $h_c = kf_c$ be the number of harmful actions.

Frequency and harmed queries. If $m_c \ge z^* \ge 1$, some action is harmful, so $\mathbf{1}[m_c \ge z^*] \le h_c = kf_c \le Kf_c$. A query loses evidence only if some action is harmful, so $r_c \le \mathbf{1}[h_c \ge 1] \le Kf_c$. If $h_c \ge 1$, at least one query loses evidence, so $r_c \ge 1/|\mathcal{Q}_c|$ and $f_c \le 1 \le Qr_c$; if $h_c = 0$ both sides vanish. If $m_c \ge z^*$, one action alone harms z^* distinct queries, so $r_c \ge z^*/Q$ and $\mathbf{1}[m_c \ge z^*] \le (Q/z^*)r_c$. Taking expectations over conversations gives the first three inequalities.

E-Tail. The exact empirical CVaR places weight $\min\{1, 1/((1-\eta)k)\}$ on the largest capped severity and averages values no larger than it, so $\min\{1, 1/((1-\eta)k)\}\,\tilde{m}_c \le \phi_c \le \tilde{m}_c$ with $\tilde{m}_c = \min\{m_c, B\}/B$. If $m_c \ge z^*$, then $\tilde{m}_c \ge \tilde{z}$, which gives the lower bound on ρ_{E}. If $m_c < z^*$, then $\phi_c \le (z^* - 1)/B$; otherwise $\phi_c \le 1$. Hence $\phi_c \le \mathbf{1}[m_c \ge z^*] + (1 - \mathbf{1}[m_c \ge z^*])(\tilde{z} - 1/B)$, since z^* is an integer, and $\rho_{\mathrm{E}} \le \rho_{\mathrm{M}}$ is the concavity argument of Appendix B.5.

MIX-CVaR. Under the mixture F_λ, the probability of a positive capped severity is $\mathbb{E}_c[f_c] = \rho_{\mathrm{F}}$, and every positive value lies in $[1/B, 1]$. The upper $(1-\eta)$ tail therefore contains positive mass $\min\{\rho_{\mathrm{F}}, 1-\eta\}$ with values in $[1/B, 1]$ and zeros elsewhere, which gives both bounds on ρ_{M}.

Equality. Each bound is attained by a distribution in which every conversation is identical: (i) K actions, one with blast radius z^* and the others harmless, attains $\rho_{\mathrm{C}} = K\rho_{\mathrm{F}}$ and, when $(1-\eta)K \ge 1$, the lower bound on ρ_{E}; (ii) one action harming exactly z^* of Q queries attains $\rho_{\mathrm{C}} = (Q/z^*)\rho_{\mathrm{H}}$; (iii) K actions of which one harms all Q queries attains $\rho_{\mathrm{H}} = K\rho_{\mathrm{F}}$; (iv) harmful actions that all harm the same single query attain $\rho_{\mathrm{F}} = Q\rho_{\mathrm{H}}$; (v) a single action with blast radius $z^* - 1$ attains $\rho_{\mathrm{E}} = \rho_{\mathrm{C}} + (1 - \rho_{\mathrm{C}})(\tilde{z} - 1/B)$ with $\rho_{\mathrm{C}} = 0$, and adding conversations with one action of blast radius B attains it for any ρ_{C}; (vi) identical conversations attain $\rho_{\mathrm{E}} = \rho_{\mathrm{M}}$; and harmful actions with blast radius 1 or at least B attain the lower or upper bound on ρ_{M}. □

When $(1-\eta)K \le 1$, E-Tail equals the mean capped maximum and bounds CAT only up to the factor B/z^*.

Raw blast radius.

Proof of the last claim of Proposition 1. Fix a frequency $a \in (0,1)$, a tail level $\eta \ge 1 - a$, and a gap $g > 0$. For $U \sim \mathrm{Unif}(0,1)$ and $r = \lceil g \rceil + 1$, set $Z_A = \mathbf{1}[U \le a]$ and $Z_B = r\mathbf{1}[U \le a]$. Both variables are nonnegative integers and satisfy $\Pr(Z > 0) = a$. For a nonnegative loss, use the exact upper-tail functional

$$\mathrm{CVaR}_\eta(Z) = \frac{1}{1-\eta}\int_\eta^1 F_Z^{-1}(p)\,dp, \qquad F_Z^{-1}(p) = \inf\{z : \Pr(Z \le z) \ge p\}.$$

Since $\eta \ge 1 - a$, $F_{Z_A}^{-1}(p) = 1$ and $F_{Z_B}^{-1}(p) = r$ for almost every $p \in (\eta, 1)$. Thus the CVaRs equal 1 and r, and their difference is $r - 1 \ge g$. □

These laws can be realized by conversations with one evicting decision: with probability a it destroys a source needed by respectively one or r future queries; otherwise it removes only unneeded memory.

A.2 Attribution Drift

Proof of Proposition 2. Take two decision times $t_1 < t_2$, three memories eligible at both, and three queries after t_2, each requiring a different memory. Use the episode-complete horizon. Scores at t_1 are $(.35, .50, .50)$, and all scores at t_2 are $.25$. The trigger and scores satisfy Assumptions 1–2 of Appendix A.4.

At $\lambda = .20$ there are no evicting decisions and the maximum is zero by convention. At $\lambda = .30$, only t_2 evicts, removing all three memories and giving severity sample $\{3\}$. At $\lambda = .40$, t_1 removes m_1 and t_2 removes m_2, m_3, giving $\{1, 2\}$. Cumulative evicted sets are nested at every time, and the number of distinct harmed queries is three at both latter thresholds. Nevertheless, the maximum follows $0, 3, 2$, and exact empirical CVaR_η falls from 3 to 2 between the latter thresholds for $\eta \in [1/2, 1)$. □

Proof of Proposition 2(ii)–(iii). Under M1–M2 each entry m has a fixed list of eligible decisions and scores, and it is removed under λ at the first eligible decision $\tau_m(\lambda)$ whose score is below λ. Hence τ_m is nonincreasing in λ, and an entry removed under λ_1 is removed no later under λ_2. Let q be counted by τ at λ_1, so some source m of q has $\tau_m(\lambda_1) = \tau < t_q$. If any such source still has $\tau_m(\lambda_2) = \tau$, then q is counted by τ at λ_2. A query can therefore leave τ only if every source removed at τ under λ_1 has $\tau_m(\lambda_2) < \tau$, which gives (ii); a drifted query can still be counted again when another of its sources is first removed at τ under λ_2. If each entry's scores do not decrease across its eligible decisions, its first eligible score is its minimum, so $\tau_m(\lambda)$ is either that first decision or undefined and never moves earlier. Then $\Delta_\tau = 0$ for every τ, each Z_τ is nondecreasing, and so are its maximum and CAT. For E-Tail and MIX-CVaR, a new harmless action enlarges the within-conversation sample or the mixture and can lower the average over the upper tail; one-shot review on LoCoMo shows 16 such decreases of E-Tail. □

Empirical check. `scripts/check_attribution_drift.py` compares adjacent thresholds on the 117-point LoCoMo grid and the main LongMemEval grid. No query leaves an action without drift (11,784 and 1,612 action comparisons); 18 and 6 drifted queries are counted again through another source. All 78 and 49 decreases of a conversation's largest blast radius coincide with drift. Truncating each entry's records to its first eligible decision, which is one-shot review, removes every such decrease; on LoCoMo, E-Tail still decreases 16 times through dilution alone, and 10 of its 131 decreases under the paper's scorer involve no drift.

In general, $\sum_\tau Z_\tau$ need not equal $\sum_q \ell_q$: a query may lose different sources at different decisions. Equality in this construction follows from its one-source-per-query design.

A.3 Splitting Compactions

Proposition 5 (Splitting)**.** *Use the full-conversation horizon, and replace an action E_τ by actions $E_{\tau_1}, \ldots, E_{\tau_r}$ that partition E_τ, with $\tau \le \tau_1 < \cdots < \tau_r$ in the same window and no query arriving in $[\tau, \tau_r]$. Then every ℓ_q and every window blast radius are unchanged, while $\max_i Z_{\tau_i} \le Z_\tau \le \sum_i Z_{\tau_i}$.*

The harmed-query rate and window-level targets are therefore invariant to splitting, whereas action-level targets reward it.

Proof of Proposition 5. No query arrives in $[\tau, \tau_r]$, so every query that arrives after τ also arrives after τ_r, and for such a query the entries evicted before its arrival are the same before and after the split, because $\bigcup_i E_{\tau_i} = E_\tau$. Queries arriving before τ see neither action. Hence every ℓ_q is unchanged. All actions lie in the window of τ and remove the same union of entries, so each window loses the same evidence for the same queries, and every window blast radius is unchanged. A future query counted by Z_{τ_i} requires an entry of $E_{\tau_i} \subseteq E_\tau$ and arrives after τ_r, hence after τ, so

it is counted by Z_τ; this gives $\max_i Z_{\tau_i} \le Z_\tau$. Conversely, a query counted by Z_τ requires an entry of some E_{τ_i} and is counted by that Z_{τ_i}, which gives $Z_\tau \le \sum_i Z_{\tau_i}$. □

A.4 Trajectory-Level Monotone Coupling

M1–M2 enable the exact shared sweep; history-dependent policies use full per-threshold replay.

Assumption 1 (M1: policy-independent scores)**.** *$s_\tau(m)$ depends only on the data stream, m's ingestion time, and τ, not on the eviction history. (Semantic, recency, and re-mention scores qualify; retrieval-hit counts do not.)*

Assumption 2 (M2: policy-independent trigger)**.** *The set of decision times does not depend on λ, and eligibility for eviction depends only on an entry's ingestion time through a fixed protection rule. (Periodic triggers qualify; capacity triggers do not: aggressive early eviction silences later triggers.)*

Lemma 1 (Trajectory-level monotone coupling)**.** *Under Assumptions 1–2 and the adapter of Eq. (1), for $\lambda_1 \le \lambda_2$ the cumulative evicted sets are nested at every time t: $\mathrm{Ev}_{\le t}(\lambda_1) \subseteq \mathrm{Ev}_{\le t}(\lambda_2)$. Consequently every query-level witness, the total harm, and the compression ratio are nondecreasing in λ.*

Proof of Lemma 1. Fix $\lambda_1 \le \lambda_2$ and induct over the common decision times supplied by M2. Both cumulative evicted sets are initially empty. Suppose they are nested immediately before decision t, and let $m \in E_t(\lambda_1)$. If m was already evicted under λ_2, it is already in the larger cumulative set. Otherwise it survives under both policies. Eligibility depends only on its ingestion time and the common protection rule, so it is eligible under λ_2. M1 gives the same score on both trajectories, and $s_t(m) < \lambda_1 \le \lambda_2$ implies that λ_2 also evicts it. All entries previously removed under λ_1 remain in the larger cumulative set by the induction hypothesis. Hence nestedness holds after t, and between decisions neither set changes.

For each query, intersecting the nested cumulative sets before its arrival with the fixed source set $\mathcal{S}_q$ preserves inclusion. Equation (2) then gives monotonicity of ℓ_q and of $\sum_q \ell_q$. Compression is monotone because the number of removed entries is monotone and the ingested stream is fixed. □

A.5 Critical Thresholds and Exact One-Pass Sweep

Lemma 2 (Critical thresholds and exact grid replay)**.** *Under Assumptions 1–2 and Eq. (1), let (t_k, v_k) be the strict prefix-minimum records of entry m's scores over all its eligible decision times on the undeleted input stream. Then t_k increases and v_k strictly decreases. With $\min \varnothing = +\infty$, simultaneously for every threshold: (i) m is evicted iff $\min_k v_k < \lambda$, at the earliest t_k with $v_k < \lambda$; (ii) for $m \in \mathcal{S}_q$, the source-query pair is harmed iff*

$$c(q, m) := \min\{v_k : t_k < t_q\} < \lambda,$$

and its responsible decision is the earliest such crossing, which is nonincreasing in time as λ grows; (iii) one chronological pass constructing these records gives an index from which every grid replay and its harm summaries can be recovered exactly.

Proof. M1 supplies scores independent of eviction history, including for entries retained on the undeleted reference stream. M2 and the fixed age rule supply common eligible times. An entry survives through t precisely when every eligible score up to t is at least λ. The first score below λ must be a strict prefix-minimum record, which proves (i), including the no-eligible-time case.

For (ii), the first crossing must occur strictly before t_q. Increasing λ enlarges the set of crossing records, so its earliest member cannot move later. This argument attributes harm to a source-query pair, not uniquely to a query with several sources.

For (iii), record each entry's ingestion time and prefix minima in chronological order. At a grid point, recover each deletion time using (i), form the nonempty sets E_τ, and count each future query once per decision when at least one of its sources is in E_τ, respecting the chosen horizon. Compute ℓ_q by an OR over its source-query witnesses. These operations recover both empty-decision conventions and all summaries of Section 5.2. □

The shared index supplies deletion times for each grid point; aggregation then yields the conversation-level losses used for calibration.

B CERTIFICATE CONSTRUCTION AND STATISTICAL GUARANTEES

Fix λ and draw i.i.d. calibration conversations as in Section 5.1. We construct the three pointwise bounds, then establish threshold selection and the relation between tails.

B.1 CAT: EXACT BINOMIAL UPPER BOUND

For the Bernoulli losses in Eq. (6), $K = \sum_i L_i \sim \mathrm{Bin}(n, \rho_{\mathrm{C}}(\lambda))$. For $k < n$, the Clopper–Pearson limit solves

$$\sum_{j=0}^{k} \binom{n}{j} U^j (1-U)^{n-j} = \delta; \qquad U(n) = 1.$$

Equivalently, $U(k)$ is the $(1-\delta)$ quantile of $\mathrm{Beta}(k+1, n-k)$.

Proof of Theorem 1 for CAT. For the true Bernoulli mean p, let $G_p(k) = \Pr_{\mathrm{Bin}(n,p)}(K \le k)$. Monotonicity of $G_p(k)$ in p implies $\{U(K) < p\} \subseteq \{G_p(K) \le \delta\}$. The latter event has probability at most δ: it consists of a lower tail of the binomial distribution whose cumulative probability does not exceed δ. The endpoint cases $p = 0, 1$ also satisfy the claim. Thus $\Pr(p \le U(K)) \ge 1 - \delta$. □

For zero events, $U = 1 - \delta^{1/n}$.

B.2 E-TAIL: A BOUNDED-MEAN CERTIFICATE

For nonempty severities sorted in decreasing order, $z_{[1]} \ge \cdots \ge z_{[m]}$, let $w = (1-\eta)m$ and $k = \lfloor w \rfloor$. The exact empirical CVaR is

$$\widehat{\mathrm{CVaR}}_\eta(z_{1:m}) = \frac{1}{w}\Big(\sum_{j=1}^{k} z_{[j]} + (w-k)\, z_{[k+1]}\Big), \tag{9}$$

where the last term vanishes when $w = k$. If $w \le 1$, it equals $z_{[1]}$. In the running example of Section 5.2 with $B = 10$, the conversation's E-Tail summary is .30 at $\lambda = .30$ and .20 at $\lambda = .40$. Apply it to capped, normalized severities within each conversation to obtain $X_i = \phi_{c_i}(\lambda) \in [0, 1]$, taking zero for no evictions. Each conversation receives equal weight.

Proof of Theorem 1 for E-Tail. The X_i are i.i.d. with mean $\mu = \rho_{\mathrm{E}}(\lambda)$. The betting construction of Waudby-Smith & Ramdas (2024) supplies, for each candidate mean m, a nonnegative capital process $K_t(m)$ starting at one and satisfying the supermartingale property when m is the true mean. Bets are predictable, depending only on earlier observations. Ville's inequality gives $\Pr_\mu(\sup_{t \le n} K_t(\mu) \ge 1/\delta) \le \delta$. Let

$$\mathcal{C}_n = \{m \in [0,1] : \sup_{t \le n} K_t(m) < 1/\delta\}, \qquad U_{\mathrm{E}} = \sup(\mathcal{C}_n \cup \{0\}).$$

On the complementary event $\mu \in \mathcal{C}_n$, and hence $\mu \le U_{\mathrm{E}}$. The one-sided WSR upper endpoint therefore has coverage at least $1 - \delta$. □

Implemented betting process and inversion. For fixed n, `ucb.py` uses

$$\hat{\mu}_t = \frac{\frac{1}{2} + \sum_{i=1}^{t} X_i}{t+1}, \quad v_t = \frac{\frac{1}{4} + \sum_{i=1}^{t} (X_i - \hat{\mu}_i)^2}{t+1}, \quad v_0 = \tfrac{1}{4},$$

$$a_t = \sqrt{\frac{2\log(1/\delta)}{n v_{t-1}}}, \qquad \nu_t(m) = \min\left\{a_t, \frac{.75}{\max\{1-m, 10^{-9}\}}\right\},$$

$$K_t(m) = \prod_{i=1}^{t} [1 + \nu_i(m)(m - X_i)].$$

Predictability follows from v_{t-1}; every factor is at least .25 and has conditional expectation one at the true mean. Reject m when $\max_{t \le n} \log K_t(m) \ge \log(1/\delta)$. Each factor is nondecreasing in m, so bisection locates the rejection boundary. Thirteen steps leave width at most 2^{-13}; returning the upper endpoint preserves coverage. If $m = 1$ is accepted, return one.

B.3 MIX-CVaR: Rockafellar–Uryasev Dual

Construction. For a fixed finite grid $\Gamma \subset [0,1]$, put $g_\beta(c) = \int (z-\beta)_+ \, d\hat{F}_\lambda^{(c)}(z) \in [0, 1-\beta]$. For $\beta < 1$, apply WSR to $g_\beta/(1-\beta) \in [0,1]$ at failure level $\delta/|\Gamma|$ and multiply its endpoint by $1-\beta$ to obtain $U_\beta(\lambda)$; set $U_1 = 0$. Then

$$U_{\mathrm{MIX}}(\lambda) = \min\left\{1,\ \min_{\beta\in\Gamma}\left[\beta + \frac{U_\beta(\lambda)}{1-\eta}\right]\right\} \tag{10}$$

is a valid upper bound, as shown below. With fixed relative slack and an adequate dual grid, the variance-adaptive route requires

$$n \gtrsim \frac{\log(|\Gamma|/\delta)}{(1-\eta)\alpha}, \tag{11}$$

up to further logarithmic factors; the zero-event floor of Appendix C.1 has the same dependence on α and $1-\eta$.

For F supported on $[0,1]$, including discrete distributions, Rockafellar & Uryasev (2002) gives

$$\mathrm{CVaR}_\eta(F) = \min_{\beta\in[0,1]}\left\{\beta + \frac{\mathbb{E}_F[(Z-\beta)_+]}{1-\eta}\right\}.$$

On a pre-specified finite grid C, let $g_\beta(c) = \int (z-\beta)_+ \, d\hat{F}_\lambda^{(c)}(z) \in [0, 1-\beta]$. Linearity gives $\mathbb{E}_c[g_\beta(c)] = \mathbb{E}_{F_\lambda}[(Z-\beta)_+]$. For $\beta < 1$, apply WSR to the range-normalized variable $h_\beta = g_\beta/(1-\beta) \in [0,1]$ at level $\delta/|\Gamma|$, and let $U_\beta = (1-\beta)U(h_\beta)$. At $\beta = 1$, $g_1 \equiv 0$, so set $U_1 = 0$. This uses each stop-loss column's exact support.

Proof of Theorem 1 for MIX-CVaR. With probability at least $1-\delta$, every U_β is valid simultaneously, by a union bound. On this event,

$$\begin{aligned}\min_{\beta\in\Gamma}\left\{\beta + \frac{U_\beta}{1-\eta}\right\} &\geq \min_{\beta\in\Gamma}\left\{\beta + \frac{\mathbb{E}_c[g_\beta(c)]}{1-\eta}\right\} \\ &\geq \min_{\beta\in[0,1]}\left\{\beta + \frac{\mathbb{E}_{F_\lambda}[(Z-\beta)_+]}{1-\eta}\right\} = \mathrm{CVaR}_\eta(F_\lambda).\end{aligned}$$

The risk is at most one, so clipping the bound at one preserves validity and gives Eq. (10). □

B.4 Fixed-Sequence Validity

Proof of Proposition 4. Condition on any independently chosen policy family, grid, and test order. If all candidates have risk at most α, the result is immediate. Otherwise let λ^* be the first candidate in this fixed order with $\rho(\lambda^*) > \alpha$. Returning a threshold with risk above α requires that the procedure first pass λ^*. Thus

$$\{\rho(\hat{\lambda}) > \alpha\} \subseteq \{U(\lambda^*) \leq \alpha < \rho(\lambda^*)\}.$$

The pointwise upper-confidence-bound property bounds the latter event's probability by δ. The same conclusion holds after averaging over any independent randomness used to choose the policy family or order. No ordering of the risks or independence among threshold tests is required. □

The fixed sequence may leave safe thresholds after the first failure unselected, but every tested bound can use the full failure level δ. Non-contiguous selection instead uses level $\delta/|\Lambda|$ for each bound. Separate guarantees for thresholds selected separately for several targets allocate δ across those targets.

Joint targets at one threshold. To bound two risks ρ_1, ρ_2 at the same deployed threshold, let a candidate pass only when $U_1(\lambda) \leq \alpha_1$ and $U_2(\lambda) \leq \alpha_2$, where each U_j satisfies Eq. (5) at the full level δ, and run the same fixed sequence. Then

$$\Pr\left\{\rho_1(\hat{\lambda}) \leq \alpha_1 \text{ and } \rho_2(\hat{\lambda}) \leq \alpha_2\right\} \geq 1-\delta.$$

Proof. Let λ^* be the first candidate in the fixed order that violates at least one target, and let j index a violated target, so $\rho_j(\lambda^*) > \alpha_j$. Returning any jointly unsafe threshold requires passing λ^*, which requires $U_j(\lambda^*) \le \alpha_j < \rho_j(\lambda^*)$. Since λ^* and j are fixed by the population, this event has probability at most δ. □

No split of δ is needed, because a jointly unsafe selection always requires one specific bound to err. The two bounds may be computed from the same calibration conversations. The same argument covers any finite number of targets.

Several ordered paths. Let a policy family consist of J paths fixed before calibration, each an ordered list of candidates, such as one threshold grid per retention scorer. Run fixed-sequence testing on each path with bounds at level δ/J and let $\mathcal{C}$ be the union of the passing prefixes. Then $\Pr\{\exists\, \pi \in \mathcal{C} : \rho(\pi) > \alpha\} \le \delta$, so any rule for choosing the deployed candidate from $\mathcal{C}$, including one that uses the calibration data, keeps risk at most α with probability at least $1 - \delta$.

Proof. On path j, an unsafe candidate enters $\mathcal{C}$ only if the first unsafe candidate of that path passes, which has probability at most δ/J by the argument for Proposition 4. A union bound over the J paths gives the claim. □

B.5 E-Tail Never Exceeds MIX-CVaR

E-Tail averages each conversation's empirical CVaR, whereas MIX-CVaR applies CVaR to the equal-conversation mixture of decision severities. For each β, the objective $\beta + \mathbb{E}_F[(Z-\beta)_+]/(1-\eta)$ is affine in F, so $\mathrm{CVaR}_\eta(F)$, their minimum, is concave in F. Jensen's inequality then gives

$$\rho_{\mathrm{E}}(\lambda) = \mathbb{E}_c\big[\mathrm{CVaR}_\eta(\hat{F}_\lambda^{(c)})\big] \le \mathrm{CVaR}_\eta\big(\mathbb{E}_c[\hat{F}_\lambda^{(c)}]\big) = \rho_{\mathrm{M}}(\lambda),$$

where a conversation without evictions contributes the point mass at zero to both sides. A MIX-CVaR certificate at α therefore also certifies E-Tail at α, but not conversely, and the gap can be large.

Harmless actions dilute the mixture tail. MIX-CVaR draws one evicting action uniformly within each conversation, so additional actions that harm no query add mass at zero and lower the mixture tail even when the harmful actions are unchanged. CAT is unaffected, because it uses each conversation's largest blast radius. E-Tail is affected only through the number of evicting actions m in its tail fraction $(1-\eta)m$. A threshold that deletes many unneeded entries can therefore look safer under MIX-CVaR than under CAT, which is the tail analogue of the frequency denominator effect in Section 6.1.

B.6 Anytime-Valid Calibration

Proposition 6 (Anytime-valid calibration)**.** *Let conversations arrive i.i.d. and, for each λ, let $U_n(\lambda)$, the bound after n conversations, be nonincreasing in n with $\Pr\{\exists n : \rho(\lambda) > U_n(\lambda)\} \le \delta$. Let $\hat{\lambda}_n$ be the fixed-sequence selection from U_n. Then $\hat{\lambda}_n$ is nondecreasing in n, and $\Pr\{\exists n : \rho(\hat{\lambda}_n) > \alpha\} \le \delta$.*

Proof of Proposition 6. Because each $U_n(\lambda)$ is nonincreasing in n, a threshold that passes after n conversations passes after every later one, so the passing prefix of the fixed order, and with it $\hat{\lambda}_n$, can only grow. Let λ^* be the first candidate in the fixed order with $\rho(\lambda^*) > \alpha$. An unsafe deployment at any time requires λ^* to pass at that time, that is, $U_n(\lambda^*) \le \alpha < \rho(\lambda^*)$ for some n, which has probability at most δ by assumption. □

Construction. For losses $x_i \in [0,1]$ with mean μ, candidate mean θ, and predictable bets $\nu_i = \min\{b_i, .75/(1-\theta)\}$ with $b_i = \sqrt{2\log(1/\delta)/(\hat{\sigma}_{i-1}^2\, i\log(i+1))}$, the capital $W_n(\theta) = \prod_{i\le n}(1 + \nu_i(\theta - x_i))$ is a nonnegative martingale at $\theta = \mu$. Rejecting θ once $\sup_{j\le n} W_j(\theta) \ge 1/\delta$ and setting $U_n = \inf\{\theta :\ \theta \text{ rejected after } n \text{ conversations}\}$ gives, by Ville's inequality, $\Pr\{\exists n : \mu > U_n\} \le \delta$, and U_n is nonincreasing. Unlike the fixed-sample WSR bound of Theorem 1, the bets do not use the final sample size. At $n = 30$ it differs from the fixed-sample WSR bound by at most .031.

C SAMPLE COMPLEXITY

C.1 ZERO-EVENT FLOOR

Proof of Proposition 3. Because U sees each conversation only through its empirical distribution of capped severities, which is the point mass at zero whenever no action harms a query, every such conversation yields the same input, and a sample with no harm is equivalent to n copies of one such conversation x_0. Fix x_0 and a conversation x_1 with a single evicting action of blast radius at least B. For $p \in [0, 1]$, let P_p draw x_1 with probability p and x_0 otherwise. Under P_p, CAT and E-Tail equal p, because x_1 has $\max_\tau Z_\tau \geq B \geq z^*$ and $\phi_{x_1} = 1$, and the mixture puts mass p on capped severity 1 and the rest on 0, so MIX-CVaR equals $\min\{1, p/(1-\eta)\}$. Let ρ_p denote the target under P_p and let U_0 be the bound on the sample of n copies of x_0. If $U_0 < \rho_p$ for some p with $(1-p)^n > \delta$, then $\Pr_{P_p}\{\rho_p > U\} \geq (1-p)^n > \delta$, which contradicts validity. Hence $U_0 \geq \rho_p$ for every $p < 1 - \delta^{1/n}$, and taking the supremum gives the two bounds. □

The floor is attained by CAT, whose Clopper–Pearson limit with zero events equals $1 - \delta^{1/n}$. For MIX-CVaR at $n = 30$ and $\delta = .1$, the floor is .148, .296, and .739 at $\eta = .5$, .75, and .9, against .369, .738, and 1 for the Bonferroni-corrected dual.

D ASSUMPTION AUDIT AND EFFICIENT REPLAY ROBUSTNESS

Full replay supports scores that depend on retained memory and triggers that depend on current capacity. Sections D.1–D.2 examine these two cases.

D.1 A HISTORY-DEPENDENT SCORER

The synthetic audit has 48 topics with redundant memory copies. Its topic-availability loss records a query when the last pre-query copy is removed, attributing harm to that decision. A bonus for the last surviving topic copy creates history dependence. The copy count at each decision uses only entries ingested before it, so the scorer uses no future information. Full replay computes exact risks over 2,000 conversations. CAT uses $n = 50$, $z^* = 2$, $\alpha = .2$, $\delta = .1$, and 500 i.i.d. with-replacement repetitions, all non-abstaining.

boost (M1-violation strength)	0 (compliant)	0.15	0.30
full replay: unsafe-selection rate	.012	.008	.028
full replay: mean $\hat{\lambda}$	.247	.352	.445
shortcut: loss misclassification	0%	2.6%	5.2%
shortcut: unsafe-selection rate	.012	.002	.000
shortcut: mean $\hat{\lambda}$	.247	.332	.385

Table 2: M1 audit with topic-availability loss. Unsafe selections use the exact full-replay pool risk. Shortcut rows calibrate on frozen-score losses; misclassification compares loss cells with full replay.

As the boost grows, full-replay unsafe-selection rates stay below δ, while the frozen-score shortcut increasingly misclassifies losses. Correct replay thus supports calibration with policy-dependent state.

D.2 CAPACITY TRIGGERS AND M2

Earlier deletion can suppress later capacity triggers. We check 66 increasing threshold pairs in each of ten LoCoMo conversations; nestedness violations rise from 0% under periodic triggers to 37.1% under capacity triggers across these 660 pairs. Capacity triggers also violate query-witness monotonicity in 24.4% of pairs. Calibration uses five $8/2$ folds and full replay at all 59 thresholds for both triggers, giving 1,180 stored profiles. CAT uses $z^* = 5$, $\alpha = .3$, and $\delta = .1$; both triggers pass all five folds. Trigger, protection, and risk settings are collected in Appendix G.

trigger	drops (compr. / harm)	passing folds	$\hat{\lambda}$ (folds)	test cat. (folds)	time
periodic	0 / 0	5/5	.03,.03,.03,.20,.03	0,0,0,.5,0	36.08 s
capacity	10 / 7	5/5	.18,.18,.18,.21,.18	0,0,0,.5,0	200.14 s

Table 3: Full-replay LoCoMo calibration. Drops count decreases in compression and pooled query harm; test catastrophe rates use two conversations per fold. Times are wall-clock measurements from the recorded run.

Each selected point has zero calibration events, giving $U = 1 - .1^{1/8} = .25010579$. The next grid point has one event and $U = .40624549$, so fixed-sequence testing stops. The table pools query harm over 1,531 questions; equal-conversation harm has six drops, using adjacent changes below -10^{-12}.

E Extended Empirical Analysis of Blast Radius

The subsections follow the main text. Appendix E.1 reports the main LongMemEval setting in full; Appendix E.2 describes the four retention scorers of Figure 2a; Appendix E.3 tests Proposition 2; Appendices E.4 and E.5 give the validity results behind Section 6.2; Appendix E.6 evaluates anytime-valid calibration; Appendix E.7 replaces annotation with retrieval logs; Appendix E.8 varies the construction; Appendix E.9 replaces the lexical scorer with sentence embeddings; and Appendix E.10 applies the certificates to memories extracted by Mem0.

E.1 Main LongMemEval Setting

Construction. The adapter call `load_longmemeval_session_disjoint(seed=1, timing="native")` joins questions that share an answer session, packs the components into 50 streams, keeps only answer sessions, and fires each question after the last retained session dated on or before its `question_date`. The streams contain 10,398 turns (136–300 per stream) and 479 queries (9–10 per stream); 256 queries fire later than under the evidence-time rule of Appendix G.1. Scoring, the 25-turn period, 20-turn protection, and the threshold grid $\{.02, \ldots, .60\}$ follow Appendix G.2. Table 4 lists every certified target; single-target rules that abstain need more calibration streams than 30 at that tolerance. MIX-CVaR ($\eta = .9$, $B = 10$) abstains on every split at $\alpha = .30$ and .45: with 30 streams its dual bound is 1 already at the first threshold, although the population MIX-CVaR stays at or below .261 on the whole grid; by Proposition 3, no such bound falls below .739 on a harm-free sample of 30 streams. Streams have at most 12 evicting actions, so at $\eta = .95$ each stream's E-Tail value is its largest capped severity. Averaged over the 20 splits, every certified risk stays below its tolerance on held-out streams, while the other risks remain free; the joint rule bounds both at the same δ.

E.2 Four Retention Scorers

Figure 2a compares four retention scorers, with weights $(w_{\text{sem}}, w_{\text{rec}}, w_{\text{freq}})$ of $(.45, .30, .25)$ (base), $(.15, .70, .15)$ (recency-led), $(.70, .15, .15)$ (semantic-led), and $(.20, .20, .60)$ (frequency-led), fixed before any result was inspected. At 40% of memory deleted, the 50-stream harmed-query rates are 16.2, 15.2, 21.8, and 14.5%, CAT ($z^* = 2$) rates 25.1, 34.7, 46.5, and 32.7%, and CAT ($z^* = 3$) rates 5.8, 13.9, 7.1, and 7.1% for the four scorers, so the scorer that minimizes one risk differs from the one that minimizes another. Over scorer $\times$ threshold, each scorer's path is tested at level $\delta/4$ and the rule deploys the certified pair with the largest calibration deletion (`run_policy_family_2d.py`). The harmed-query rate at .25 then deploys the recency-led scorer on 17 of 20 splits and CAT ($z^* = 3$) at .20 the semantic-led or base scorer on 15. CAT ($z^* = 2$) at .30 deletes 29.0% of memory, against 29.2% with the base scorer alone, and 13.3% of test streams contain a multi-query eviction, against 18.3%.

Target	α	Mean $\hat{\lambda}$	Deleted	Harmed queries	CAT, z^*=2	CAT, z^*=3
Frequency	.05			abstains on all splits		
Frequency	.10	.184	0.3%	0.1%	0.0%	0.0%
Frequency	.15	.254	2.6%	0.9%	0.8%	0.2%
Frequency	.20	.439	20.2%	6.7%	10.2%	4.0%
Frequency	.25	.461	29.4%	10.4%	17.0%	5.5%
Frequency	.30	.476	35.8%	14.5%	22.5%	7.5%
Frequency	.40	.518	52.9%	26.1%	48.3%	17.5%
Harmed-query rate	.05			abstains on all splits		
Harmed-query rate	.10	.251	0.6%	0.4%	0.0%	0.0%
Harmed-query rate	.15	.429	16.5%	5.4%	7.8%	2.8%
Harmed-query rate	.20	.460	28.7%	9.9%	17.0%	5.0%
Harmed-query rate	.25	.482	38.3%	15.9%	24.3%	6.8%
Harmed-query rate	.30	.500	45.9%	20.7%	37.5%	9.5%
CAT (z^*=2)	.05			abstains on all splits		
CAT (z^*=2)	.10	.404	8.1%	2.8%	2.8%	0.8%
CAT (z^*=2)	.20	.427	15.8%	5.1%	8.5%	3.0%
CAT (z^*=2)	.30	.461	29.1%	10.3%	18.3%	5.5%
CAT (z^*=2)	.50	.493	43.0%	18.4%	29.3%	7.3%
CAT (z^*=3)	.05			abstains on all splits		
CAT (z^*=3)	.10	.427	16.4%	5.7%	8.2%	3.5%
CAT (z^*=3)	.20	.489	40.9%	17.3%	30.5%	5.8%
CAT (z^*=3)	.30	.518	53.0%	25.9%	48.5%	15.5%
E-Tail	.05			abstains on all splits		
E-Tail	.10	.251	0.6%	0.4%	0.0%	0.0%
E-Tail	.15	.442	21.4%	6.9%	10.5%	4.8%
E-Tail	.20	.487	40.5%	16.9%	28.3%	5.8%
E-Tail	.25	.524	55.2%	28.0%	56.2%	17.2%
E-Tail	.30	.587	73.4%	42.9%	80.0%	33.5%
CAT (z^*=2) + Harmed-query rate	.10 + .10	.251	0.6%	0.4%	0.0%	0.0%
CAT (z^*=2) + Harmed-query rate	.10 + .15	.404	8.1%	2.8%	2.8%	0.8%
CAT (z^*=2) + Harmed-query rate	.10 + .20	.404	8.1%	2.8%	2.8%	0.8%
CAT (z^*=2) + Harmed-query rate	.10 + .25	.404	8.1%	2.8%	2.8%	0.8%
CAT (z^*=2) + Harmed-query rate	.10 + .30	.404	8.1%	2.8%	2.8%	0.8%
CAT (z^*=2) + Harmed-query rate	.20 + .10	.251	0.6%	0.4%	0.0%	0.0%
CAT (z^*=2) + Harmed-query rate	.20 + .15	.424	14.8%	4.9%	8.0%	2.2%
CAT (z^*=2) + Harmed-query rate	.20 + .20	.427	15.8%	5.1%	8.5%	3.0%
CAT (z^*=2) + Harmed-query rate	.20 + .25	.427	15.8%	5.1%	8.5%	3.0%
CAT (z^*=2) + Harmed-query rate	.20 + .30	.427	15.8%	5.1%	8.5%	3.0%
CAT (z^*=2) + Harmed-query rate	.30 + .15	.429	16.5%	5.4%	7.8%	2.8%
CAT (z^*=2) + Harmed-query rate	.30 + .20	.459	28.3%	9.6%	16.5%	4.8%
CAT (z^*=2) + Harmed-query rate	.30 + .25	.461	29.1%	10.3%	18.3%	5.5%
CAT (z^*=2) + Harmed-query rate	.30 + .30	.461	29.1%	10.3%	18.3%	5.5%
CAT (z^*=3) + Harmed-query rate	.10 + .10	.251	0.6%	0.4%	0.0%	0.0%
CAT (z^*=3) + Harmed-query rate	.10 + .25	.427	16.4%	5.7%	8.2%	3.5%
CAT (z^*=3) + Harmed-query rate	.20 + .10	.251	0.6%	0.4%	0.0%	0.0%
CAT (z^*=3) + Harmed-query rate	.20 + .25	.478	36.8%	14.9%	22.7%	6.8%

Table 4: Every certified target in the main LongMemEval setting ($\delta = .1$), means over the 20 calibration/test splits of Section 6.3. Test columns: memory deleted, queries that lose evidence, and streams with an action harming at least z^* queries.

E.3 Selection Under Non-Monotone and Monotone Risk

Non-monotone population. The 50 streams define a population whose risk at each threshold is the stream mean. Each repetition draws n streams with replacement and selects a threshold either by fixed-sequence testing or by taking the largest threshold whose bound passes. Table 5 counts selections whose population risk exceeds α. On the non-monotone CAT ($z^* = 3$) curve, the largest passing threshold is unsafe up to 6.1% of the time, and more often than fixed-sequence selection at every sample size where a threshold is returned; both stay within $\delta = 10\%$ in these configurations.

Target	α	Monotone	$n = 30$	$n = 50$	$n = 100$	$n = 200$
CAT (z^*=2)	.10	yes	1 / 4	0 / 0	2 / 2	0 / 0
CAT (z^*=2)	.20	yes	8 / 28	8 / 30	10 / 19	6 / 12
CAT (z^*=3)	.05	no	0 / 0	26 / 61	10 / 22	10 / 21
CAT (z^*=3)	.10	no	1 / 19	0 / 14	0 / 11	0 / 20
E-Tail	.05	no	0 / 0	0 / 0	0 / 0	0 / 0
E-Tail	.10	no	0 / 0	0 / 0	0 / 0	0 / 0
Harmed-query rate	.05	yes	0 / 0	0 / 0	0 / 0	1 / 1
Harmed-query rate	.10	yes	0 / 0	0 / 0	0 / 0	0 / 0

Table 5: Unsafe selections out of 1,000 repetitions on the 50-stream LongMemEval population (fixed-sequence / largest passing threshold). "Monotone" reports whether the population risk is nondecreasing on the grid.

A non-monotone step on LoCoMo. Over the 117 thresholds of Figure 2b, CAT ($z^* = 5$) falls at 9 steps and E-Tail at 38, while the harmed-query rate never falls. Under the fusion scorer, raising λ from .450 to .455 raises the harmed-query rate from 18.24% to 18.39% while CAT ($z^* = 5$) falls from 70% to 60% and E-Tail from 66.8% to 65.2% (Figure 2b).

One-shot review. One-shot review judges each entry only at its first eligible decision, so its score cannot decay afterwards and, by Proposition 2(iii), CAT is nondecreasing in λ; its 50-stream CAT curve is monotone, whereas the paper's scorer gives a non-monotone one. We compare fixed-sequence selection with taking the largest passing threshold on the 20 splits and on 1,000 resamples of 30 streams (`RandomState(41)`). Under one-shot review the two rules select the same threshold on every split and resample for CAT ($z^* = 2$) at .10 and .20 and for the harmed-query rate at .15, deleting 3.4, 5.6, and 4.6% of memory with 10, 30, and 0 unsafe resamples; fixed-sequence testing loses nothing here when risk is monotone. Under the paper's decaying scores, the largest passing threshold for CAT at .20 deletes 19.3% instead of 15.8% but is unsafe in 27 resamples instead of 11. Decaying scores thus delete more at the same tolerance (15.8% against 5.6%) and pay for it with attribution drift. Source: `outputs/one_shot_review`.

E.4 Synthetic Coverage and Calibration Cost

Design and interpretation. In Figure 3, CAT uses $z^* = 4$ and $\alpha = .10$, E-Tail $\eta = .95$ and $\alpha = .10$, and MIX-CVaR $\eta = .9$ and $\alpha = .40$, for $n \in \{10, 50, 100, 200, 500\}$ on a log axis; CAT and MIX-CVaR use the short pool and E-Tail the 100-decision pool, so that each conversation's 5% tail spans several actions; MIX-CVaR uses $\alpha = .40$, consistent with its $1/((1-\eta)\alpha)$ cost. Figure 3 checks three aspects of selection: (a) the fraction of all repetitions that select an unsafe threshold, counting abstention as safe; (b) the fraction that return a threshold; and (c) the mean deployed threshold, assigning zero on abstention, relative to the largest safe grid threshold $\lambda^\star = \max\{\lambda \in \Lambda : \rho_{\text{pool}}(\lambda) \le \alpha\}$.

The pools retain strong dependence among actions within each conversation. The 40,000-conversation short pool supplies reference risks on $\{.02, .04, \ldots, .80\}$. Calibration samples conversations i.i.d. with replacement. An *unsafe selection* returns a threshold whose reference risk exceeds α; *abstention* returns no threshold because the first candidate fails. Unsafe-selection frequency divides by all repetitions, and abstention is reported separately. The CAT and MIX reference curves on the short pool and the E-Tail reference curve on the 100-decision pool are nondecreasing on the grid, so the largest safe threshold $\lambda^\star$ of Figure 3(c) is also the last threshold before the first unsafe one and is reachable by fixed-sequence testing.

Results and oracle proximity. More calibration conversations make selection less conservative: CAT returns a threshold in every repetition at $n = 200$, MIX-CVaR moves from complete abstention at $n = 100$ to always returning one at $n = 200$ (at $\alpha = .40$ the floor of Proposition 3 excludes only $n \leq 56$; the rest of the gap comes from the Bonferroni-corrected dual), and at $n = 500$ the mean deployed thresholds reach 77.2% (CAT), 95.5% (E-Tail), and 82.5% (MIX-CVaR) of each target's largest safe threshold (Figure 3b,c).

The short pool has at most twelve evicting decisions per conversation, so at $\eta = .95$ its E-Tail loss is the capped maximum. The long pool (20,000 conversations, 100 decision times; active conversations average 84–93 evictions across thresholds) lets each conversation's CVaR average its 4–5 largest capped severities (Table 7).

target (α)	$n = 10$	$n = 50$	$n = 100$	$n = 200$	$n = 500$
CAT (.10)	abstains	6/1000 (26.7%)	4/1000 (0.8%)	5/1000	0/1000
FREQ (.25)	abstains	9/1000	23/1000	32/1000	28/1000
MIX (.40)	abstains	abstains	abstains	0/1000	0/1000
E-Tail (.05)	abstains	abstains	8/1000 (56.8%)	5/1000 (20.2%)	14/1000 (1.1%)
E-Tail (.10)	abstains	4/1000 (21.2%)	25/1000 (0.7%)	32/1000	8/1000
E-Tail (.25)	abstains	66/1000	40/1000	39/1000	21/1000

Table 6: Short-pool unsafe selections out of 1,000 repetitions; parentheses give the abstention rate when nonzero. "Abstains" means no policy is returned in any repetition. FREQ certifies the harmful-decision frequency. CAT uses $z^* = 4$; MIX uses $\eta = .9, B = 10$; E-Tail uses $\eta = .95, B = 10$. Columns $n = 10, 50, 200$ use repetition seed 13; $n = 100, 500$ use seed 43.

	$\alpha = .05$ ($\lambda^\star$=.12)	$\alpha = .10$ ($\lambda^\star$=.18)	$\alpha = .25$ ($\lambda^\star$=.34)
$n = 10$	abstains	abstains	abstains
$n = 50$	abstains	0/1000; .096 (abst. 0.2%)	0/1000; .286
$n = 100$	0/1000; .047 (abst. 1.5%)	3/1000; .141	3/1000; .309
$n = 200$	0/1000; .081	20/1000; .162	3/1000; .319
$n = 500$	0/1000; .098	19/1000; .172	1/1000; .325

Table 7: Long-episode E-Tail ($\eta = .95$, 100 decision times): unsafe count per 1,000 repetitions; mean threshold among returned policies; nonzero abstention. $\lambda^\star$ is the pool oracle.

Joint coverage. Adding each conversation's harmed-query rate to the short pool and requiring CAT ($z^* = 4$, $\alpha = .10$) and a harmed-query rate of at most .15 at one threshold, jointly unsafe selections stay at or below 9 of 1,000 repetitions for $n \in \{50, 100, 200, 500\}$ (all runs abstain at $n = 10$). The CAT bound fails first on this pool, so the joint rule selects the CAT-only threshold, whereas the harmed-query bound alone is jointly unsafe in 879 to 1,000 repetitions once $n \geq 50$.

E.5 Decision-Level Calibration

A decision-level bound treats all eviction actions from the sampled conversations as independent draws. On the synthetic short pool, the Clopper–Pearson bound for the pooled share of actions with blast radius at least $z^* \in \{2, 4\}$ misses its target in at most 6.6% of 1,000 repetitions for $n \in \{30, 100, 300\}$, and fixed-sequence selection with it is unsafe in at most 32 repetitions. Within-conversation dependence in this generator is therefore too weak to break the decision-level bound. In the main LongMemEval setting, decision-level miscoverage reaches 10.2% ($n = 100$), close to δ; on merged streams with native question timing it reaches 15.6% (Appendix E.8).

E.6 Anytime-Valid Calibration

Each of 1,000 runs draws the 50 LongMemEval streams one at a time with replacement, up to 200, and after every draw deploys the fixed-sequence selection from the time-uniform betting bounds of Appendix B.6 on the grid $\{.30, \ldots, .60\}$ (`RandomState(2027)`). Deployments only move to larger thresholds, as Proposition 6 states. For CAT ($z^* = 2$) at .20, the harmed-query rate at .15, and

E-Tail at .15, 2.5%, 0.7%, and 0% of runs ever deploy a threshold whose population risk exceeds the tolerance at any time along the whole path. The deployment reaches $\lambda = .40$, the threshold that CAT at .10 certifies with 30 fixed streams, after 14 streams for CAT in every run, and after a median of 23 (range of the middle 80%: 21–25) and 22 (20–24) streams for the other two. For comparison, the fixed-sample Clopper–Pearson bound falls below .20 after 11 event-free streams, since $.8^{11} < .1 \leq .8^{10}$. Continuing to annotate relaxes the deployment: mean deletion grows from 15.9, 16.9, and 21.5% after 30 streams to 24.1, 31.4, and 45.6% after 200.

E.7 Retrieval-Log Proxy Evidence

A deployment can keep an uncompressed shadow copy of memory during calibration and log, for each query, the top-k entries returned by its own retriever. Treating these entries as required evidence removes annotation. We use the QA pipeline's lexical retriever and calibrate on proxy risks, then evaluate every selected threshold under the annotated evidence (Table 8). No proxy-selected threshold exceeds its tolerance under the annotated 50-stream risk. Proxy evidence marks more entries as required than the annotations do, so it registers more harm and usually certifies smaller thresholds. At $k = 5$ the proxy covers at least one annotated entry for 68.9% of queries.

k	Target	α	Proxy $\hat{\lambda}$	Gold $\hat{\lambda}$	Unsafe (gold)	Deleted (proxy)	Deleted (gold)
5	CAT (z^*=2)	.10	.401	.404	0/20	7.1%	8.1%
5	CAT (z^*=2)	.20	.409	.427	0/20	9.7%	15.8%
5	CAT (z^*=3)	.10	.431	.427	0/20	17.0%	16.4%
5	Harmed-query rate	.15	.408	.429	0/20	9.3%	16.5%
5	Harmed-query rate	.25	.447	.482	0/20	23.4%	38.3%
5	E-Tail	.15	.412	.442	0/20	10.3%	21.4%
10	CAT (z^*=2)	.10	.362	.404	0/20	2.4%	8.1%
10	CAT (z^*=2)	.20	.393	.427	0/20	5.8%	15.8%
10	CAT (z^*=3)	.10	.408	.427	0/20	9.3%	16.4%
10	Harmed-query rate	.15	.393	.429	0/20	5.6%	16.5%
10	Harmed-query rate	.25	.423	.482	0/20	14.1%	38.3%
10	E-Tail	.15	.401	.442	0/20	7.5%	21.4%

Table 8: Calibration with retrieval-log proxy evidence (top-k entries from an uncompressed shadow memory). Proxy evidence contains at least one annotated evidence entry for 68.9% ($k = 5$) and 82.9% ($k = 10$) of queries. "Unsafe (gold)" counts proxy-selected thresholds whose annotated 50-stream risk exceeds α.

E.8 Sensitivity: Shared Distractors

LongMemEval-S draws each question's haystack from a common distractor corpus. Grouping ten questions per stream and keeping every haystack session gives 50 merged streams of about 4,900 turns in which 1,209 of 1,225 stream pairs share a session, so these streams are not independent calibration units. Both constructions below keep distractor sessions and use native question timing.

Merged streams. The largest blast radius is four queries, and over the 50 streams the harmed-query rate rises from 39.5% to 96.6% between $\lambda = .26$ and .60 while CAT ($z^* = 2$) moves through 42, 42, 20, 44, and 36% at $\lambda = .26, .30, .40, .50, .60$. Frequency calibration at .15 then deletes 95.1% of memory with 81.1% of queries harmed while 5.86% of eviction actions are harmful; loose tail tolerances fail in the same way, with CAT ($z^* = 3$) at .30 deleting 98.3% with 94.4% harmed and E-Tail at .20 deleting 99.2% with 96.7% harmed. On this non-monotone CAT curve at $\alpha = .35$, taking the largest passing threshold is unsafe in 115, 81, 72, and 44 of 1,000 repetitions for $n = 30, 50, 100, 200$, whereas fixed-sequence selection is unsafe in at most 11. A decision-level Clopper–Pearson bound for the pooled share of actions harming two or more queries misses its target in 14.5% ($n = 30$) and 15.6% ($n = 100$) of repetitions at $\lambda = .40$, where the conversation-level bound for CAT misses in 3.7% and 8.1%.

Session-disjoint streams with distractors. Keeping every distractor session in the first stream that uses it gives 50 streams of 2,851–5,015 turns that share no session, with native question timing.

Under the paper's scorer, 47.7% of minimum scores fall in $[.25, .26)$ and the largest blast radius is 6. Frequency calibration at .15 and .20 deletes 90.3 and 99.1% of memory while 68.1 and 97.2% of queries lose evidence and 5.7 and 9.9% of eviction actions are harmful: most evictions remove only distractor turns, so the harmful share stays small while query harm grows. Meanwhile CAT ($z^* = 2$) at .10 and .20 and the harmed-query rate at .15 and .25 stop below the cluster and delete 3.9%. Long distractor turns thus bring back the score cluster of the lexical scorer (configuration A of `outputs/distractor_embed_pilot`).

E.9 Embedding Scorer

The embedding scorer, fixed before any result was inspected, scores entry m at decision τ as $.6 \max\{0, \max_k \langle e_m, e_k \rangle\} + .4 \exp(-a \ln 2/100)$, where e are L2-normalized all-MiniLM-L6-v2 sentence embeddings, k ranges over the last 10 ingested turns, and a is the entry's age in turns; the 25-turn period and 20-turn protection are unchanged (`embed_lme_turns.py`, `sweep_embed.py`). Scores depend only on the stream, so Assumption 1 holds and the one-pass sweep is exact.

Main setting. The largest .01-wide bin holds 4.4% of minimum scores, so thresholds move deletion smoothly. Table 9 lists the selections on the 20 splits: certifying only the harmed-query rate at .25 leaves 28.7% of test streams with an action harming two or more queries, and adding CAT at .20 lowers this to 8.2% at 25.5% deletion, the same pattern as the lexical scorer in Table 1.

Table 9: Certified selections with the embedding scorer in the main LongMemEval setting ($\delta = .1$), means over the 20 splits; columns as in Table 1.

Certified target	α	Mean $\hat{\lambda}$	Deleted	Harmed queries	CAT, z^*=2	CAT, z^*=3
CAT (z^*=2)	.10	.201	12.1%	2.4%	2.5%	0.0%
CAT (z^*=2)	.20	.245	25.5%	6.1%	8.2%	1.2%
CAT (z^*=3)	.10	.255	28.9%	8.6%	11.2%	3.3%
CAT (z^*=2) + harmed queries	.20 + .25	.245	25.5%	6.1%	8.2%	1.2%
Harmed-query rate	.15	.241	24.0%	5.4%	6.0%	0.5%
Harmed-query rate	.25	.293	42.1%	16.5%	28.7%	4.0%
Frequency	.15	.178	9.5%	1.8%	1.5%	0.0%
Frequency	.20	.233	21.1%	4.7%	5.0%	1.0%
E-Tail	.15	.249	26.6%	6.7%	9.0%	2.2%
E-Tail	.20	.296	42.8%	17.0%	30.5%	5.0%

Streams with distractors. These streams average 3,850 turns, against a 100-turn recency half-life, so old entries lose their recency term, distractor entries match little of the recent context, and 50.2% of minimum scores fall in $[0, .01)$. The smallest threshold, .02, already deletes 62.7% of memory with 36.9% of queries harmed, so CAT, the harmed-query rate, and their joint rule abstain on every split. Frequency at .15 deletes 89.0% of memory with 63.0% of queries harmed, as under the lexical scorer (Appendix E.8). Streams of this length call for a scorer whose recency scale grows with the stream.

E.10 Mem0 Memories

Construction. Mem0 (version 2.2.1; Chhikara et al., 2025) reads the 50 main-setting streams session by session and extracts memories with DeepSeek-V4.1-Flash at temperature zero with thinking disabled, embedding them with all-MiniLM-L6-v2 in a local vector store (`mem0_build.py`). Its `add` call only creates memories, so each memory has one source session and enters the store when that session ends. The 900 sessions yield 10,636 memories; the provider's content filter rejected one session and three produced no memory. Extraction used 899 calls with 10.7M input and 0.68M output tokens, 377 through the DeepSeek API and 522 through DashScope, which serves the same model.

Eviction and evidence. Memories are scored with the embedding scorer of Appendix E.9 against the raw turns of the stream, with the same period, protection, and grid $\{.02, \ldots, .80\}$; scores de-

pend only on the stream, so the one-pass sweep is exact. The memory store is built once without eviction, so eviction does not feed back into later extraction. Two labelings, fixed before any result was inspected, define required evidence (`mem0_analyze.py`): *provenance* maps each annotated evidence turn to the memory from its session whose embedding is closest to the turn's, and *retrieval* takes the 5 memories closest to the question among those created before it, as Mem0's search on the uncompressed store returns. Four queries whose evidence sessions produced no memory are dropped under provenance, leaving 475.

Results. The largest .01-wide bin holds 4.4% of minimum scores, so deletion moves smoothly with the threshold, and the largest blast radius is 4 queries under provenance and 7 under retrieval. Table 10 lists the selections on the paper's 20 splits. Under provenance, CAT at .10 and .20 deletes 15.0% and 27.8% of memories with test CAT risks of 2.2% and 8.0%, while the harmed-query rate alone at .25 deletes 44.3% and leaves 27.0% of test streams with an action harming two or more queries. Retrieval labels mark more memories as required and certify smaller thresholds, with the same ordering of rules.

Table 10: Certified selections on Mem0 memories ($\delta = .1$), means over the 20 splits under the two evidence labelings: deleted memories, harmed queries, and test streams with an action harming at least two queries.

		Provenance			Retrieval		
Certified target	α	Deleted	Harmed	CAT	Deleted	Harmed	CAT
CAT (z^*=2)	.10	15.0%	1.7%	2.2%	8.2%	3.3%	3.8%
CAT (z^*=2)	.20	27.8%	6.8%	8.0%	12.5%	5.2%	6.8%
CAT (z^*=3)	.10	31.2%	8.8%	12.8%	13.9%	6.5%	8.3%
CAT (z^*=2) + harmed queries	.20 + .25	27.8%	6.8%	8.0%	12.5%	5.2%	6.8%
Harmed-query rate	.15	25.2%	5.7%	4.0%	11.7%	4.3%	4.8%
Harmed-query rate	.25	44.3%	16.2%	27.0%	23.4%	15.0%	25.0%
Frequency	.15	18.1%	1.9%	2.2%	4.5%	1.3%	0.5%
Frequency	.20	26.9%	6.6%	6.8%	7.3%	2.5%	2.5%
E-Tail	.15	30.5%	8.2%	12.5%	12.5%	5.3%	6.8%

F End-Task Validation and Robustness

F.1 QA in the Main Setting

Section 6.4 answers the 479 LongMemEval queries of the main setting without eviction, at every threshold in $\{.40, \ldots, .49\}$, each certified on some split by a rule of Table 12, and at .50 and .60. Table 12 scores each split's certified threshold on that split's 20 test streams. We also answer the 1,531 LoCoMo queries without eviction and at $\lambda \in \{.25, .30\}$. A lexical retriever passes the top 20 live entries to a DeepSeek-V4.1-Flash reader, and Qwen3.7-Max judges each answer against the gold answer; retrieval, prompts, and parsers follow Appendix G.4. Accuracy loss grows with the number of queries whose evidence is deleted (Table 11). Unharmed queries change in both directions at similar rates, and queries whose evidence the baseline retriever already missed almost never change (Table 13), so the correct-to-wrong changes concentrate where retrievable evidence was deleted. On LoCoMo, overall accuracy is unchanged ($+0.3$ and $+0.1$ points) and harmed queries change 4:1 at .25 and 7:5 at .30.

G Reproducibility and Implementation Details

Table 14 collects the settings linking data, replay, calibration, and QA evaluation.

G.1 Datasets, Query Timing, and Conversation Units

LoCoMo. The adapter reads the authors' official `locomo10.json` release (MD5 `03e2a21d45726cd5ad29602c4f9998e0`), retains categories 1–4, and removes dan-

Table 11: QA accuracy by eviction threshold. Changes are against no eviction with 95% bootstrap intervals over streams (LongMemEval) or conversations (LoCoMo). Flips count correct-to-wrong and wrong-to-correct answers among queries that lost evidence.

Benchmark	λ	Accuracy	Change (points)	Harmed queries	Flips
LongMemEval	no eviction	.489	—	0	—
LongMemEval	.40	.484	-0.4 [-1.9, $+1.1$]	11	2 : 1
LongMemEval	.41	.474	-1.5 [-3.1, 0.0]	16	3 : 0
LongMemEval	.42	.486	-0.2 [-2.1, $+1.7$]	21	7 : 2
LongMemEval	.43	.466	-2.3 [-4.4, -0.4]	25	8 : 2
LongMemEval	.44	.466	-2.3 [-4.3, -0.4]	31	10 : 2
LongMemEval	.45	.474	-1.5 [-3.4, $+0.4$]	37	9 : 2
LongMemEval	.46	.463	-2.5 [-4.4, -0.4]	46	15 : 1
LongMemEval	.47	.453	-3.5 [-5.8, -1.3]	61	18 : 2
LongMemEval	.48	.451	-3.8 [-6.5, -0.8]	75	23 : 2
LongMemEval	.49	.434	-5.4 [-8.5, -2.3]	81	25 : 3
LongMemEval	.50	.428	-6.1 [-9.0, -3.1]	97	31 : 2
LongMemEval	.60	.301	-18.8 [-23.3, -14.4]	218	94 : 1
LoCoMo	no eviction	.432	—	0	—
LoCoMo	.25	.435	$+0.3$ [-0.4, $+1.2$]	82	4 : 1
LoCoMo	.30	.433	$+0.1$ [-0.7, $+1.0$]	161	7 : 5

Table 12: QA at the thresholds certified by eight rules of Table 1 and Table 4. Each split scores the threshold its rule certifies on that split's 20 test streams; changes are against no eviction, averaged over the 20 splits with the range over splits, and flips sum correct-to-wrong and wrong-to-correct changes among harmed test queries over the splits.

Certified target	α	Mean $\hat{\lambda}$	Change (points)	Flips
CAT (z^*=2)	.10	.404	-0.8 [-3.7, $+1.0$]	22 : 4
CAT (z^*=2)	.20	.427	-1.2 [-4.2, $+1.5$]	59 : 13
CAT (z^*=3)	.10	.427	-1.8 [-4.1, $+1.0$]	62 : 9
CAT (z^*=2) + harmed queries	.20 + .25	.427	-1.2 [-4.2, $+1.5$]	59 : 13
Harmed-query rate	.15	.429	-2.1 [-4.2, $+1.0$]	65 : 13
Harmed-query rate	.25	.482	-4.8 [-9.2, 0.0]	181 : 19
Frequency	.20	.439	-1.8 [-3.7, $+1.0$]	68 : 13
E-Tail	.15	.442	-2.1 [-3.7, $+0.5$]	68 : 13

Benchmark	λ	Harmed	Net loss (points)	Retrievable (flips)	Baseline miss (flips)	Unharmed flips
LongMemEval	.40	11	9.1 [-25.0, 38.5]	8 (2:1)	3 (0:0)	6:5
LongMemEval	.43	25	24.0 [0.0, 47.6]	22 (8:2)	3 (0:0)	9:4
LongMemEval	.50	97	29.9 [19.6, 39.8]	87 (31:2)	10 (0:0)	8:8
LongMemEval	.60	218	42.7 [35.6, 49.6]	197 (93:1)	21 (1:0)	8:11
LoCoMo	.25	82	3.7 [0.0, 6.4]	53 (4:0)	29 (0:1)	11:19
LoCoMo	.30	161	1.2 [-2.6, 4.7]	105 (6:3)	56 (1:2)	17:21

Table 13: Paired answer changes among harmed queries. Net loss is correct-to-wrong minus wrong-to-correct changes per harmed query, with 95% cluster-bootstrap intervals (10,000 resamples). Retrievable queries had gold evidence in the baseline top 20; flips are correct-to-wrong : wrong-to-correct.

gling or empty evidence. Conversations have 369–689 turns. The log records nine dangling references and 455 excluded questions, leaving 1,531 queries. Sources retain dialogue identifiers; sessions follow numerical order. Each query fires before the session following its last usable evidence session, or at $T + 1$ when that session is final.

LongMemEval. The evaluated snapshot is the benchmark authors' September 2025 cleaned LongMemEval-S release, `longmemeval_s_cleaned.json` (MD5 `9b009af4084c6207e9d41aa0dcb6506c`, 277,383,467 bytes). The authors' release note links its upstream row-level change log. We use this JSON without local rewriting; the adapter performs every transformation. Evidence comes from `has_answer` turns in annotated answer sessions, and removing 21 evidence-free `_abs` questions leaves 479 queries. Sessions are sorted by date and ID, with midnight for dates lacking a clock time, and sources use session ID and turn index.

The main setting calls `load_longmemeval_session_disjoint(seed=1, timing="native")`. It joins questions that share an answer-session ID, assigns each connected component to one of 50 balanced streams, keeps only answer sessions, and fires each question after the last retained session dated on or before its `question_date`, never before the evidence-time rule below; streams share no session. With `distractors=True`, each distractor session is kept in the first stream that uses it (Appendix E.8). `build_index_lme_disjoint_native.py` caches the sweep index.

Under the evidence-time rule, a query fires before the session following its last annotated answer session, or at $T + 1$. Merged streams shuffle the 500 questions with `random.Random(1)` into 50 groups of ten and keep each haystack session once; 1,209 of 1,225 stream pairs share a haystack session.

G.2 Scoring, Triggering, and Replay

Scorer. Tokenization lowercases `[a-z']+` matches, removing words of length at most two and the 171 fixed stopwords archived in `configs/scorer_stopwords.txt`. Let W_m be an entry's word set, C_t the union over the last ten turns, t_m its ingestion time, and $r_t(m)$ its re-mention count. Then

$$s_{\text{sem}} = \frac{|W_m \cap C_t|}{|W_m|}, \quad s_{\text{rec}} = 2^{-(t-t_m)/100}, \quad s_{\text{freq}} = \min\{1, r_t(m)/3\}.$$

Fusion uses $.45s_{\text{sem}} + .30s_{\text{rec}} + .25s_{\text{freq}}$; empty W_m gives semantic score zero. Two shared content words increment a live entry's count. Scores use observed turns only.

Adapter and event order. Queries scheduled at a turn run before its observation. Scorer update, ingestion, and compaction follow. Periodic compaction runs every 25 turns; the capacity variant triggers above 100 live entries. Eligibility requires $t - t_m > 20$ and deletion requires $s_t(m) < \lambda$. Each full replay resets the store and scorer. M1–M2 permit the shared prefix-minimum index; history-dependent variants use full replay.

Tail convention. E-Tail applies the exact empirical CVaR of Eq. (9) with $\eta = .95$, $B = 10$ within conversations. Empty conversations contribute zero (a point mass for MIX).

G.3 Calibration and Statistical Settings

Compression is the fraction of memory entries deleted. Reported rates average streams (LongMemEval) or conversations (LoCoMo) equally within each split.

Splits and aggregation. All LongMemEval split experiments, including the fixed-threshold profile of Figure 4, use successive permutations from `RandomState(17)`. The first 30 indices calibrate and the last 20 test. LoCoMo test pairs are $(26, 30), (41, 42), (43, 44), (47, 48), (49, 50)$. Reported split means use each returned threshold's own test outcomes; abstention is reported separately. Standard deviations use divisor n.

Bounds and selection. The bounds in Appendix B feed the fixed-sequence selection of Section 5.4.

Experiment	Configuration
LongMemEval	Main setting: 50 session-disjoint streams, native question timing; 30 calibration and 20 test streams, 20 splits; $\delta = .1$, CAT $z^* \in \{2, 3\}$, E-Tail $\eta = .95$, $B = 10$. Merged streams use grouping seed 1.
Extensions	Four scorers (Figure 2a), grid $\{.02, \ldots, .80\}$; proxy evidence $k \in \{5, 10\}$; population resampling $n \in \{30, 50, 100, 200\}$, 1,000 repetitions.
LoCoMo	Five fixed 8/2 folds; CAT $z^* = 5$, $\alpha = .3$, $\delta = .1$.
Benchmark grids	Ascending $\{.02, .03, \ldots, .60\}$ (59 points); fine tail curve $\{.02, .025, \ldots, .60\}$ (117 points); nestedness audit $\{.05, .10, \ldots, .60\}$ (12 points).
Synthetic audits	Short pool 40,000 and long pool 20,000 conversations; grid $\{.02, .04, \ldots, .80\}$ (40 points); calibration draws i.i.d. with replacement; $n \in \{10, 50, 100, 200, 500\}$; 1,000 repetitions per configuration; $\delta = .1$.
MIX dual	$\eta = .9$, $B = 10$; stop-loss grid $\{0, .1, \ldots, 1\}$ for benchmarks and $\{0, .1, .2, .3\}$ for synthetic data.
History-dependent audit	Pool size 2,000; $n = 50$; 500 repetitions; CAT $z^* = 2$, $\alpha = .2$, $\delta = .1$; boosts 0, .15, .30.
QA resampling	Main setting: 10,000 stream or conversation resamples, `RandomState(2026)`; percentile intervals.

Table 14: Implementation settings for the benchmark, synthetic, and QA experiments.

Synthetic construction and sampling. The short pool uses seed 7, 300 turns, and 30 query times obtained by flooring equally spaced values from 60 to 300. Each query samples two distinct past entries with rank weights $r^{-1.2}$. Entry scores sum a shared $\mathrm{Unif}(-.05, .35)$ conversation factor and independent $\mathrm{Unif}(.02, .67)$ noise, clipped to $[.01, 1]$; each has one critical record at its first eligible decision. The long pool uses $T = 2{,}500$ and 100 decision times. History-dependent runs use 180 turns, 48 topics, 24 queries, a 24-turn trigger, grid $\{.06, .085, \ldots, .66\}$, and pool/calibration seeds $3/11$. All calibration draws sample conversations with replacement. Unsafe selection and abstention are defined once in Appendix E.4.

G.4 End-Task Evaluation and Compute

Retrieval and models. For each query and condition, lexical retrieval ranks live entries by question-word overlap, breaking ties by recency then source ID. The top 20 entries are truncated to 800 characters each and presented chronologically. Reader and judge run at temperature zero with thinking disabled and output limits of 256 and 8 tokens. The reader answers from supplied memories or emits `UNKNOWN`; the judge accepts semantically equivalent, noncontradictory answers via `OK`/`NO`. Empty or `UNKNOWN`-prefixed answers count as incorrect; other answers require case-insensitive `OK` with at most one terminal punctuation mark. Exact prompts and parsers are archived.

Statistical summaries. Main-setting intervals resample streams (LongMemEval) or conversations (LoCoMo) 10,000 times with `RandomState(2026)`.

Main-setting runs. The runs of Section 6.4 call the DeepSeek API with model `deepseek-v4-flash`, served as `deepseek-flash` (DeepSeek-V4.1-Flash) with one system fingerprint across all reader calls, and the DashScope API with `qwen3.7-max` as judge; answers of `UNKNOWN` are scored incorrect without a judge call. Each record stores the SHA-256 of the exact prompt, the returned model name, response identifier, fingerprint, and token counts; a run manifest stores code and snapshot hashes.

G.5 Artifact Organization and Reproduction Scope

Archived configurations and membership maps link each split to its replay profiles, selections, and QA records. The packaged source and outputs support record replay; dataset adapters regenerate streams from the named benchmark snapshots. The machine-readable `reproducibility_manifest.json` records both official dataset URLs, file sizes, MD5 and SHA-256 fingerprints, every LongMemEval exclusion, and the stopword-file checksum.