What Makes Structured Episodic Retrieval Robust to Noisy Query Cues? A Controlled Study

Anonymous ACL submission

Abstract

Structured memories for LLM agents index events by time, place, entity and topic, and answer a query by intersecting the matches on each axis. Intersection fails when an LLM extracts the events and the query cues, because one weak axis removes the right memory. We run a controlled study of retrieval rules on EPBench and LoCoMo. Hard intersection returns nothing for 63% of EPBench questions, and adding up the per-axis matches instead recovers most of the loss. How best to add them up depends on what the answer does with the retrieved set. A ranked top-K set helps questions that list events but hurts questions that ask for the most recent event, and routing those questions to a selective rule recovers the retrieval loss. Cue confidence is informative, yet as a single weight per axis it cannot change the result in most cases and leaves retrieval and answer accuracy essentially unchanged. On LoCoMo, matching the speaker, which is metadata, raises answer accuracy over dense retrieval by 7.9 points ($[6.0, 9.9]$; 7.0 for the full score), while turning the topic score into a yes-or-no indicator, or choosing the weights on other conversations, makes no detectable difference. All results carry paired bootstrap intervals, and our conclusions are limited to these two benchmarks.

1 Introduction

Long-running LLM agents must recall what happened, when, where and with whom. Dense retrieval over embeddings (Lewis et al., 2020) finds related text, but it does not represent the factors that define an episode (Tulving, 1983). Structured episodic memories add these factors as explicit axes. STEM (Chang and Tan, 2017) and its LLM-based successor ARTEM (Tan et al., 2026) store time, place, entity and content channels. They retrieve with Fusion ART (Tan et al., 2019), which requires every queried axis to reach a fixed vigilance threshold, that is, a minimum match score. Retrieval is therefore an intersection.

A cue is the value that an LLM extracts from the question for one axis, such as a date or a name. An intersection assumes that every cue is right, and with LLM-extracted events and cues this assumption fails. The content axis in particular rarely reaches a threshold such as 0.98. We ask how structured retrieval behaves under this noise and which of three changes repairs it: (i) dropping the hard intersection, (ii) grading the match instead of thresholding it, and (iii) weighting each axis by the confidence of its cue.

We compare four rules on one two-part memory. One part (Tier 2) stores the memory cells (events on EPBench, dialogue turns on LoCoMo). The other (Tier 1) holds simple index maps from a cue value, such as a date, a name or a word, to the cells that contain it; they only narrow down the candidates. The cells follow STEM and ARTEM and the index maps are ordinary inverted indexes, so we do not present the memory layout as new. We add three alternatives to their hard intersection (Gated-AND, Literal and Graded; Section 3.3), a calibrated confidence for each cue, and a controlled comparison with paired bootstrap intervals. In tables and figures we call the graded, confidence-weighted sum CIRCUIT. We report four findings:

- **The way matches are combined has to fit what the question asks for.** Keeping the K best-scoring events retrieves more of the right chapters for questions that ask for a list or an order of events and fewer for questions that ask for the most recent event (gold-chapter recall, the share of answer chapters whose event reaches the context: $+0.116$ and -0.479). The ranked set also holds later events that are not the answer, and the latest of them gets picked. Sending "latest" questions to a stricter

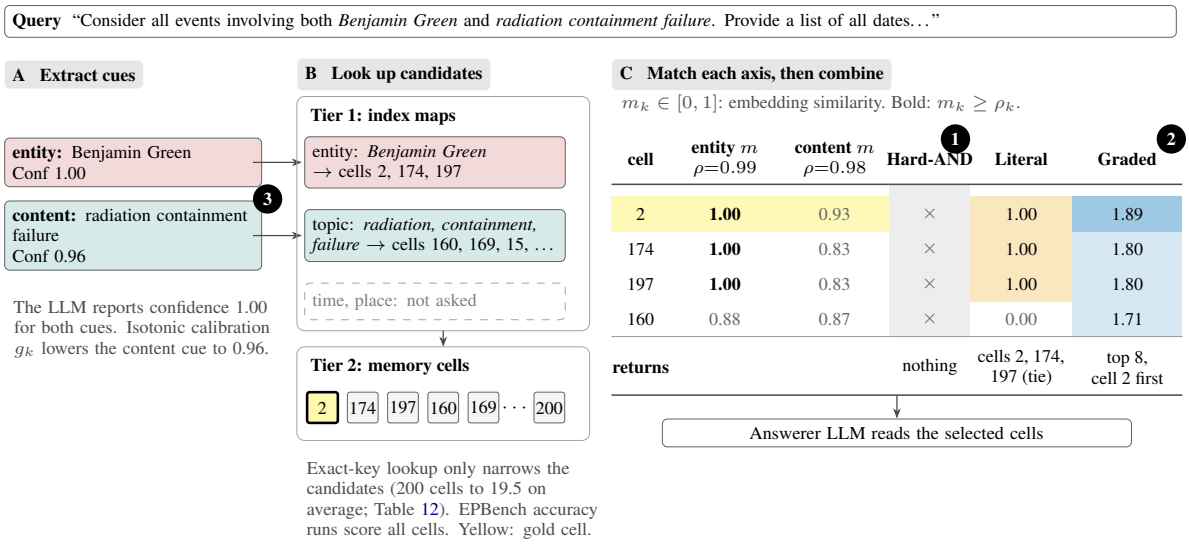


Figure 1: Retrieval rules on one EPBench question. (A) An LLM extracts cue values with a self-reported confidence, and isotonic calibration turns it into Conf_k. (B) Tier-1 index maps look up exact keys and list candidate cells; Tier 2 stores the cells. (C) Each candidate gets a graded embedding match m_k per axis, and three rules combine the matches. Marked findings: (1) Hard-AND returns nothing for 63% of EPBench questions. (2) The graded sum ranks well for list questions but hurts questions that ask for the latest event. (3) Calibrated confidence changes the retrieved cells for only 41 of 668 questions.

rule recovers the loss (§5.2).

- **Hard intersection breaks down.** It returns nothing for 63.2% of the EPBench questions, and for 49.6% of those that have an answer. Adding up the matches instead of requiring all of them (a binary confidence-weighted sum) gains +0.217 F1, and lowering the content threshold does not close the gap (§5.1).

- **Confidence tracks whether a cue is right but changes almost nothing.** Calibrated, raw and no confidence give results we cannot tell apart on both benchmarks, and one confidence number per axis cannot change the outcome in most cases (Proposition 1, §5.3). On LoCoMo, the gain over dense retrieval comes from matching the speaker's name, which needs no extracted cue, and turning the topic score into a yes-or-no indicator changes nothing (§5.4).

- **Efficiency, measured again.** An index cuts the cells to score by 90.3% on EPBench. In timed runs, scoring only the candidates is about 3× faster than scoring every cell (a flat scan), and embedding a new cue string costs more than either (§5.5).

2 Related Work

Agent memory. Most agent memories rely on dense retrieval (Lewis et al., 2020). Generative Agents (Park et al., 2023) rank memories by a fixed sum of recency, importance and relevance. These weights do not depend on the cues of the query. Graph-based systems use traversal or personalized PageRank (Gutiérrez et al., 2024, 2025), linked notes (Xu et al., 2025) or temporal graphs (Rasmussen et al., 2025). Mem0 (Chhikara et al., 2025), MemOS (Li et al., 2025) and EverMemOS (Hu et al., 2026) add lifecycle management. These systems fold extraction quality and match quality into one score. We separate the two and measure what the separation buys.

Structured episodic memory. STEM (Chang and Tan, 2017) encodes time, place, entity and content channels and recalls by Fusion ART (Tan et al., 2019). ARTEM (Tan et al., 2026) adds LLM-based extraction and is evaluated on EPBench (Huet et al., 2025). Both treat cues as hard constraints with fixed vigilance. Our Hard-AND baseline reimplements this rule, because the original engine code was not available to us.

Confidence and calibration. Neural scores are often miscalibrated (Guo et al., 2017), and verbalized LLM confidence tends to be overconfident (Xiong et al., 2024; Tian et al., 2023). Isotonic re-

gression (Zadrozny and Elkan, 2002) is a standard post-hoc remedy. We do not claim calibration as new. We test whether calibrated cue confidence changes retrieval and answers.

What is new. We follow STEM and ARTEM for the cell structure, the Fusion ART match, the vigilance thresholds, the answer and judge prompts and the EPBench protocol. New here are (i) three alternatives to their hard intersection (Gated-AND, Literal and Graded), (ii) a per-axis calibration of the extractor's confidence and its use in the score, and (iii) controlled comparisons that separate dropping the intersection, grading the match and weighting by confidence.

3 Method

A query Q constrains some axes, written $A(Q)$: time, place, entity, topic or persona (the speaker). For each such axis k, an LLM extracts a cue C_k and a self-reported confidence. Calibration turns the confidence into $\mathrm{Conf}_k \in [0, 1]$, the estimated probability that the cue is right. A memory cell e gets a graded match $m_k(e) \in [0, 1]$ on each axis, and a rule combines the matches. We use the query of Figure 1 as a running example.

3.1 Memory and retrieval flow

Figure 1 shows the design. *Tier 2* stores memory cells. A cell is one event (EPBench) or one dialogue turn (LoCoMo), with its text, extracted entities and topic, and metadata such as date, place and speaker. *Tier 1* holds index maps from a cue value to the IDs of the cells that contain it. We use one map per axis: year–month buckets for time, token inverted indices for place, entity and topic, and the speaker for persona. These maps are flat. We do not build the hierarchical time and place trees or the topic–persona graph that a fuller design could use. LoCoMo has no place information, so only the time, persona and topic maps are active there.

Algorithm 1 lists the steps. The index maps look up exact keys, such as a token or a year–month, and so only narrow the candidates. Semantic similarity enters at the next step, where each candidate gets a graded match m_k per axis, computed from embeddings (Section 3.4). A combination rule then turns the matches into the context for the answering LLM.

Algorithm 1 Retrieval rules for one query Q

1: $\{(C_k, \hat{c}_k)\}_{k \in A(Q)} \leftarrow$ LLM cue extractor(Q)
2: $\mathrm{Conf}_k \leftarrow g_k(\hat{c}_k)$ for each active axis k
3: candidates $\leftarrow \bigcup_k \mathrm{index}_k[C_k]$ (or all cells)
4: $m_k(e) \leftarrow$ match of e to C_k, for e in candidates
5: **Hard-AND:** keep e with $m_k(e) \geq \rho_k$ for all k
6: **Gated-AND:** as Hard-AND over k with $\mathrm{Conf}_k \geq \gamma$
7: **Literal:** $S(e) = \sum_k \mathrm{Conf}_k \mathbf{1}[m_k \geq \rho_k]$, keep $S{>}0$
8: **Graded:** $S(e) = \sum_k \mathrm{Conf}_k\, m_k(e)$; keep top K
9: **if** the benchmark marks Q as "latest": return the time-latest kept cell
10: **else** return all kept cells in time order

3.2 Cues and confidence

Conf_k estimates how likely a cue is right. It is attached to a different thing on each benchmark: the query's cue on EPBench and a stored memory on LoCoMo.

EPBench: confidence of a query cue. For a query Q the extractor returns a value C_k and a self-reported confidence $\hat{c}_k \in [0, 1]$ for each axis k that the question constrains. The axis being asked about is excluded. Let $Z_k = 1$ if C_k agrees with the benchmark's reference cue. We fit one isotonic regression g_k per axis on pairs $(\hat{c}_k, Z_k)$ and set $\mathrm{Conf}_k = g_k(\hat{c}_k)$. Five-fold cross-fitting ensures that no question is scored by a calibrator fitted on its own label. If all cues on an axis have the same label, there is nothing to fit and the reported confidence is used unchanged.

LoCoMo: confidence of a stored memory. The extractor reads each session and emits anchors, each with a persona, a topic and a self-reported confidence. We calibrate these confidences by leave-one-conversation-out isotonic regression. A turn therefore carries its own confidence, and different candidate turns can have different values for the same query.

3.3 Combination rules

Let ρ_k be the vigilance threshold of axis k. We compare four rules on the same candidates, cues and judge (Algorithm 1, lines 5–8):

- **Hard-AND** (the STEM/ARTEM rule) keeps e if $m_k(e) \geq \rho_k$ on every $k \in A(Q)$.
- **Gated-AND** does the same over the axes with $\mathrm{Conf}_k \geq \gamma$, or over all axes if none qualifies.
- **Literal** scores $S(e) = \sum_k \mathrm{Conf}_k\, \mathbf{1}[m_k(e) \geq \rho_k]$ and keeps every cell with $S(e) > 0$.
- **Graded** scores $S(e) = \sum_k \mathrm{Conf}_k\, m_k(e)$ and keeps the top K cells. It has no per-axis

threshold.

Graded returns cells whenever any score is positive. It is therefore empty only when the query has no usable cue, and it cannot decline to answer.

In the running example, Conf is 1.00 for the entity and 0.96 for the content. Cell 2 matches the entity with 1.00 and the content with 0.93. Hard-AND rejects it because $0.93 < 0.98$. Literal scores it $1.00 \cdot 1 + 0.96 \cdot 0 = 1.00$, and Graded scores it $1.00 \cdot 1.00 + 0.96 \cdot 0.93 = 1.89$.

Proposition 1 (when a single confidence per axis cannot change the result). Let $\mathrm{Conf}_k > 0$ for every active axis. (a) If one axis is active, or all Conf_k are equal, the graded sum ranks the cells exactly as $\sum_k m_k(e)$ does, so the graded top-K set does not depend on the confidence values. (b) The Literal result set equals $\{e : \exists k,\ m_k(e) \geq \rho_k\}$ for every positive confidence. (c) Hard-AND does not use confidence. *Proof.* (a) A common positive factor preserves the order. (b) $S(e)$ is a sum of non-negative terms, and it is positive exactly when one indicator is 1. (c) By definition. □ Confidence can therefore change a result only through Graded on queries with at least two axes of different confidence, or through the threshold γ of Gated-AND.

3.4 Instantiations

EPBench. Cells are events with time, place, entities and content. For the three text axes, m_k is the Fusion ART overlap between the cue vector and the cell vector. Both come from MiniLM embeddings (Reimers and Gurevych, 2019) and are complement-coded (each value x is paired with $1 - x$), as in STEM (formula in Appendix B). For time it is $1 - |t_Q - t_e|$ on min–max scaled dates. The thresholds are the published values $\rho = (1.0, 0.99, 0.99, 0.98)$ for time, place, entity and content. We set $\gamma = 0.75$ and $K = 8$.

LoCoMo. Cells are dialogue turns. The score of a turn e adds three terms:

$$s(e) = s_{\mathrm{den}}(e) + w_p\, s_{\mathrm{per}}(e) + w_t\, s_{\mathrm{top}}(e). \quad (1)$$

- **Dense.** $s_{\mathrm{den}}(e) = \cos(q, e)$ is the BGE-base similarity (Xiao et al., 2023) between the question and the turn.
- **Persona.** $s_{\mathrm{per}}(e) = 1$ if the turn's speaker is named in the question. It equals the calibrated persona confidence if the extractor attributes the turn to the named person, and 0 otherwise.
- **Topic.** $s_{\mathrm{top}}(e) = \mathrm{Conf}_t(e) \cos(t_Q, \tau_e)$. Here τ_e is the topic anchor stored with the turn, $\mathrm{Conf}_t(e)$ is its calibrated confidence, and t_Q is the question without stop words and speaker names.

We set $w_p = 0.2$ and $w_t = 0.15$. The earlier flat-scan accuracy run (Table 10) used 0.15 for both. The top 10 sessions by best turn score are kept, and the 50 best turns among them form the context. This additive score is the graded rule of Section 3.3 on LoCoMo. The binary variants of Section 5.4 replace $\cos(t_Q, \tau_e)$ by the indicator $\mathbf{1}[\cos \geq 0.42]$ (the repository's threshold, not tuned here) and keep the calibrated topic confidence. They leave the persona term unchanged, and they run with or without the dense term.

4 Experimental Setup

EPBench. Book 1 (Uscifi, 200 chapters, 89,736 tokens) has 668 questions of six retrieval types and three modes: all, latest and chronological. Of these, 180 have an empty gold answer and 488 are answerable. Book 2 (Udefault, 196 chapters, 102,870 tokens) has 686 questions, 180 with an empty gold answer. Events come from chapter text via Qwen2.5-32B-Instruct (Qwen Team, 2024). Cues, answers and judgments come from DeepSeek-R1-Distill-Qwen-14B (DeepSeek-AI, 2025), following the ARTEM protocol. The mode of a question is a field of the benchmark, not predicted by our system. For "latest" questions the protocol scores the time-latest retrieved event. The other two modes ask for a list or an order of events, and the protocol scores all retrieved events. The calibrator is fitted on Book 1 and applied unchanged to Book 2. EP-Bench accuracy runs score all cells. The index is used for the efficiency study and for LoCoMo.

LoCoMo. We use all ten conversations and the non-adversarial questions (Maharana et al., 2024). The 1,536 with gold turn ids form the end-to-end runs of Table 3. An earlier confidence ablation used 1,540 questions and another inference engine, which scored about 2 points higher on the same retrieval (Appendix B), so we compare variants only within a run. Qwen2.5-32B-Instruct answers with a chain-of-thought prompt and judges with a lenient grader. Appendix B lists the prompts.

Metrics and statistics. Besides F1 and accuracy we use three measures that need no judge. The *empty-context rate* is the share of questions with

nothing retrieved. *Context answer-recall* is the share of gold answer items found verbatim in the retrieved events. *Gold-chapter recall* is the share of the gold answer's source chapters whose event is in the context that the protocol scores. The judge, an LLM that grades each answer against the gold answer, gives an unparseable output for 17 to 33 of 668 questions per rule. Lenient F1 over all questions counts these as zero. We therefore report F1 on the 611 questions (634 on Book 2) that parse for every rule (the judge-valid questions), and give all-question values in Appendix C. Intervals are 95% percentile bootstrap intervals over questions (10,000 resamples), and differences are paired (Efron and Tibshirani, 1993). They do not capture LLM sampling variance, and they treat questions as independent given the memory. Resampling the 10 LoCoMo conversations instead widens the interval of the speaker gain in gold turns in context (Section 5.4, +12.0 points) to $[7.7, 16.9]$ and keeps the confidence differences within ± 2 points. EPBench has only two memories.

5 Results

5.1 Hard intersection fails when an LLM extracts the cues

Table 1 gives the main results, and Appendix C gives the full F1 table with intervals (Table 5). Hard-AND returns nothing for 63.2% of the 668 questions and for 49.6% of the 488 answerable ones. Its F1 is 0.508. Gated-AND is better (+0.066, $[0.047, 0.087]$) but is still empty for 38.9% of the answerable questions. The binary confidence-weighted sum (Literal) gains +0.217 F1 over Hard-AND ($[0.183, 0.251]$), and the graded sum gains +0.240 ($[0.194, 0.285]$). Book 2 shows the same order. Abandoning the intersection therefore accounts for most of the repair.

Relaxing the threshold does not close the gap. The original content threshold is $\rho = 0.98$. At 0.95 the retrieved sets are unchanged. At 0.90 the empty rate falls to 47.9% and F1 rises to 0.628, yet the graded sum stays ahead by 0.120 ($[0.079, 0.160]$; Figure 3c and Table 15 in Appendix C). At 0.85, 38.5% of questions are still empty. We swept only the content threshold, so we cannot exclude that another setting of all thresholds closes the gap.

5.2 Where grading helps and where it hurts

Over all questions the graded sum's F1 exceeds Literal's by only 0.023 ($[-0.016, 0.060]$). On Book 2 the difference is 0.019 ($[-0.016, 0.055]$). The aggregate hides opposite effects (Figure 4, Appendix C). On answerable questions the graded sum retrieves more gold answer items (+0.063, $[0.028, 0.097]$). It also cuts the empty-context rate from 14.5% to 1.2%. On the 142 judge-valid questions with an empty gold answer, returning nothing is correct. There the graded sum loses 0.113 F1 to Literal ($[-0.218, -0.007]$), because it always returns K cells.

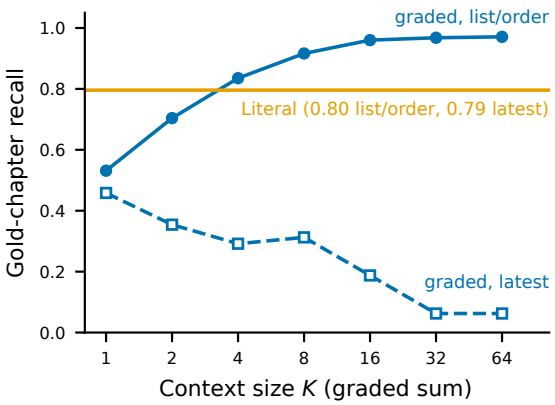


Figure 2: Gold-chapter recall of the graded sum as the context size K varies, on EPBench Book 1 answerable questions (list or order: 440; latest: 48). The orange line is Literal, which has no K. We used $K = 8$ throughout and ran this sweep afterwards.

Table 2 splits the answerable questions by question mode, number of constrained axes and cue correctness, using gold-chapter recall. We defined these splits after the main results, and a cue counts as correct if it matches the reference cue under a lenient string match. For questions that list or order events (440 answerable) the graded sum is better by 0.116 ($[0.085, 0.149]$). The gain comes from queries with a single cue axis (253 questions, +0.207). There, Hard-AND, Gated-AND and Literal coincide because each reduces to one threshold, and the graded ranking retrieves more gold events than the thresholded set. With several axes and all cues right, the two sums are indistinguishable (-0.003). With a wrong or missing cue (25 questions) the difference is not significant (-0.031, $[-0.120, 0.019]$).

The picture reverses for questions that ask for the most recent event. On the 48 answerable ones the graded sum falls 0.479 below Literal ($[-0.625, -0.333]$). End-to-end F1 agrees: -0.417 on 46 judge-valid latest questions, against $+0.116$ on 423 judge-valid list or order questions.

Rule	Confidence	EPBench Book 1 Empty (%)	F1	Recall list/order	Recall latest	Abstains (%)	LoCoMo Gold in ctx. (%)	QA acc. (%)
Dense retrieval	–	0.0	–	0.518	0.042	0.0	77.4	72.5
Hard-AND	–	63.2	0.508	0.486	0.271	100.0	–	–
Gated-AND	γ=0.75	55.2	0.574	0.549	0.667	99.4	–	–
Literal (binary sum)	calibrated	31.3	0.725	0.800	0.792	76.7	–	–
Binary sum, no dense term	calibrated	–	–	–	–	–	54.1	52.6
Graded (CIRCUIT)	calibrated	1.0	0.748	0.916	0.312	0.6	89.6	79.5
Graded	raw	1.0	–	0.921	0.292	0.6	89.6	–
Graded	none	1.0	–	0.920	0.292	0.6	89.5	–
Router: latest to Literal, else Graded	calibrated	3.1	–	0.916	0.792	8.3	–	–

Table 1: Controlled comparison of retrieval rules. EPBench Book 1 (668 questions): Empty is the share of all questions with an empty context; F1 is the lenient F1 on the 611 judge-valid questions; Recall is gold-chapter recall, without a judge, on the 440 list or order and the 48 latest answerable questions; Abstains is the share of the 180 questions with an empty gold answer that receive an empty context. Dense retrieval and the router are described in Section 5.2. LoCoMo: Gold in ctx. is the share of 1,536 questions whose context holds a gold turn (flat scan, top 10 sessions, 50 turns); QA acc. is the end-to-end accuracy on 1,536 questions from the single-session run of Table 3; the weights are w_p=0.2, w_t=0.15. The binary variant on LoCoMo binarizes the topic term and drops the dense term. Dashes: not run.

Question group	Answ.	Hard	Gated	Literal	Graded	Graded − Literal	Calibrated − constant
List or order events, at most one cue axis	253	0.691	0.691	0.691	0.898	+0.207 [+0.158, +0.258]	+0.000 [+0.000, +0.000]
List or order events, several axes, cues right	162	0.241	0.334	0.960	0.957	-0.003 [-0.025, +0.015]	-0.009 [-0.025, +0.000]
List or order events, several axes, a cue wrong or missing	25	0.000	0.497	0.866	0.835	-0.031 [-0.120, +0.019]	+0.000 [+0.000, +0.000]
Ask for the most recent event ("latest")	48	0.271	0.667	0.792	0.312	-0.479 [-0.625, -0.333]	+0.021 [+0.000, +0.062]

Table 2: Gold-chapter recall of the retrieved events on EPBench Book 1 (answerable questions only; no LLM judge), by question mode, number of constrained cue axes and cue correctness. Graded and Literal use calibrated confidence; the last column compares calibrated with constant confidence inside the graded rule. Paired 95% bootstrap intervals. "Latest" questions are scored on the most recent retrieved event.

The protocol answers a "latest" question with the time-latest retrieved event. A thresholded set holds only qualifying events, so its latest member is usually right. A ranked top-K set also holds lower-ranked events, and one of them may be later than the true answer. Figure 2 shows the interaction with K. For list or order questions, recall rises with K (0.835 at K=4, 0.971 at K=64) and passes Literal's 0.800 from K=4. For latest questions it falls from 0.312 at K=8 to 0.062 at K=32. At K=1, where one event is returned and the time rule has nothing to choose from, recall is 0.458, still below Literal's 0.792. The two question types want opposite settings of the same parameter.

A direct count supports this explanation (judge-free). Of the 48 answerable latest questions, the gold event is in the retrieved set for 38 under Literal and also for 38 under the graded sum at K=8. Literal selects it every time. The graded set holds a later, non-gold event for 21 of the 38, and the protocol selects the gold event for only 15. The other 10 questions are missed by both rules, 9 of them with a wrong or missing cue. With all cues right (14 questions), the gold event is in the graded set for 13 and is selected for 4, against 13 for Literal. The loss therefore comes from the extra later events in the set; the gold event is usually retrieved.

Routing and other baselines. Table 1 also lists dense retrieval and a router. Dense retrieval over the question text, with the same MiniLM and 8 events, reaches recall 0.518 on list or order questions, against 0.916 for the graded sum (-0.398, $[-0.441, -0.356]$). Appendix C also compares the sum without confidence, the product of the matches, the maximum, the minimum and thresholds on the graded score. The first two match the graded sum and the others fall below it, so additive and multiplicative soft aggregation both work. The router sends latest questions to Literal and all others to the graded sum. It keeps the graded recall of 0.916 on list or order questions and recovers Literal's 0.792 on latest questions. Here the question mode comes from the benchmark label. A keyword rule on the question text finds all 63 latest questions with 42 false alarms and gives the same recall. A memory abstains when it returns nothing for a question whose gold answer is empty. The router rarely does: it abstains on 8.3% of those questions, against 76.7% for Literal, so abstention remains a separate trade-off. We ran no judged F1 for the router.

Latest questions are also where the extractor

most often adds a cue that the reference does not have. Of the 63, 45 have a set of cue axes that differs from the reference: 17 add a time cue such as "the last time they were observed", 21 add a content cue, and 7 lack the entity. Only 33 of the 605 other questions (5%) differ. The extractor treats the word "last" as a constraint, so cue errors and question mode are confounded, which is why Table 2 separates them.

5.3 Cue confidence is informative but barely changes retrieval

Confidence tracks whether a cue is right. Table 8 (Appendix C) compares confidence on right and wrong cues. After calibration, 19 wrong time cues average 0.14 against 0.99 for 205 right ones. Forty-eight wrong content cues average 0.58 against 0.89. Calibration lowers the expected calibration error from 0.074 to 0.013 for time and from 0.135 to 0.069 for content. On Book 2 the Book 1 calibrator gives 0.061 to 0.005 and 0.158 to 0.065 (Table 14). Place and entity cues are right in nearly all cases, so there is little to fit.

Yet it barely changes retrieval. Table 7 (Appendix C) replaces the calibrated confidence by raw self-reports or by a constant and measures retrieval without a judge. In the graded rule, calibrated minus constant confidence changes gold-chapter recall by -0.001 ($[-0.007, 0.005]$). The retrieved events differ for only 41 of 668 questions; for raw confidence they differ for 5. Among questions with several axes and a wrong or missing cue, where confidence should matter most, the gain is $+0.019$ ($[0.000, 0.057]$, 53 answerable questions). At most 41 questions change, so a judged F1 run could move the F1 difference by at most $41/668$. We did not run one. On LoCoMo the result is the same (Table 10 in Appendix C, earlier engine). Calibrated minus no confidence is -0.2 points ($[-1.7, 1.2]$), and raw minus none is $+0.3$ ($[-1.1, 1.8]$).

Two structural reasons. A single positive confidence c_k per axis cannot change (i) the ranking of cells when one axis is active, since $S(e) = c_k m_k(e)$, nor (ii) the membership of the Literal result set. Half of the EPBench questions (349 of 668) constrain at most one axis, so (i) applies to them. Reason (ii) explains why Literal's retrieved set changes for one question only. Confidence can alter the graded ranking only for the 319 multi-axis questions, and it did so for 41 of them.

5.4 LoCoMo: what the answerer gains from structure

Table 3 gives end-to-end accuracy for seven retrieval variants that share one answerer, one judge and one session. Dense retrieval alone reaches 72.5% ($[70.2, 74.7]$). Adding the speaker match raises it to 80.4% (+7.9 points, $[6.0, 9.9]$). The full graded score, with the topic term and calibrated confidence, gives 79.5% (+7.0, $[5.0, 9.1]$; $[4.2, 10.1]$ over conversations). The topic term adds nothing to the speaker match (-0.9, $[-2.5, 0.7]$). The gain is largest on temporal questions (+11.5, $[7.2, 16.2]$) and not detectable on the 92 open-domain ones. Weights chosen on the other nine conversations give 80.6%, 1.1 points above the in-sample weights ($[-0.3, 2.5]$).

Binarizing the topic term changes little; dropping the dense term costs a lot. The binary variants turn only the topic term into a yes-or-no indicator (Section 3.4). With the dense term kept, they give 79.6%, 0.1 points above the graded sum ($[-1.4, 1.7]$). Because the topic term adds nothing beyond the speaker match even when graded, this null result has little power: it does not show that graded and binary matching are equivalent in general, and we did not binarize the persona or dense terms. Dropping the dense term costs 27.0 points for the binary variant (52.6%, $[-29.8, -24.4]$ against binary with dense) and 35.1 points for the graded terms (44.4%, $[-37.9, -32.3]$ against the full score). The 26.5-point drop of the earlier binary variant (Table 10) is therefore due to the missing dense term. Accuracy tracks the evidence: when the context holds a gold turn, the answerer is right for 85–86% of the questions in the variants with dense retrieval, and otherwise for 28–36%.

Independent judge. Because the answerer and the judge are one model, claude-sonnet-5 re-graded dense, full and held-out weights on 1,032 questions of seven conversations with the same prompt (Table 11, Appendix C). It agrees with the Qwen judge on 92.9% of the 3,096 answers ($\kappa = 0.81$) and is stricter by 4.9 points. The gain of the full score over dense retrieval is +7.1 points ($[4.5, 9.6]$) under it, against +5.0 ($[2.6, 7.6]$) under Qwen on the same questions. We did not re-grade the other four variants, and no human checked either judge.

Evidence reaching the answerer. Table 9 (Appendix C) measures retrieval directly, without an

Variant	Gold in ctx. (%)	Single (841)	Multi (282)	Temp. (321)	Open (92)	Overall [95% CI]	Minus dense [95% CI]
Dense only	77.4	78.5	68.8	67.3	46.7	72.5 [70.2, 74.7]	–
Dense + speaker	89.5	85.3	77.0	80.4	46.7	80.4 [78.4, 82.4]	+7.9 [+6.0, +9.9]
Dense + speaker + topic (graded, calibrated)	89.6	84.3	75.5	78.8	50.0	79.5 [77.5, 81.5]	+7.0 [+5.0, +9.1]
same, weights chosen on the other nine conversations	89.6	85.6	78.0	78.5	50.0	80.6 [78.6, 82.5]	+8.1 [+6.0, +10.2]
Dense + speaker + binary topic term	89.6	83.0	77.0	80.1	55.4	79.6 [77.6, 81.6]	+7.2 [+5.1, +9.2]
Graded speaker and topic terms, no dense	43.5	44.5	51.1	41.4	33.7	44.4 [41.9, 46.9]	-28.1 [-31.1, -25.1]
Binary topic term, no dense	54.1	51.1	59.9	52.0	45.7	52.6 [50.1, 55.1]	-19.9 [-22.7, -16.9]

Table 3: LoCoMo end-to-end accuracy (%), 1,536 questions, ten conversations; one run per variant in one session (vLLM, greedy), with Qwen2.5-32B-Instruct as answerer and judge. Context: top 10 sessions, 50 turns; w_p=0.2, w_t=0.15 unless stated. Gold in ctx.: share of questions whose context holds a gold turn. Last column: paired difference to dense retrieval in points (95% bootstrap interval over questions).

LLM. Dense retrieval puts a gold turn into the context for 77.4% of the 1,536 questions. Adding the speaker match raises this to 89.5% (+12.0 points, $[10.2, 13.9]$). The topic match alone adds 1.4 points ($[0.2, 2.5]$) and nothing once the speaker match is present (+0.1, $[-0.8, 1.0]$). Calibrated, raw and no confidence agree within 0.1 points. The speaker match compares the name in the question with the speaker recorded for each turn. The speaker comes from metadata and needs no extracted cue, so confidence cannot explain the gain. The gain is positive in all 10 conversations. Choosing the weights on nine conversations and testing on the tenth gives 89.6% ($[88.0, 91.1]$), close to the in-sample 89.8%. Without the dense term, retrieval falls to 54.1%. A named speaker matches 55% of the turns on average, and 98.7% of the questions name one. The indicator therefore ties for half of the turns, and the tie order decides.

5.5 Index-first retrieval and its cost

Table 12 (Appendix C) compares index-first retrieval with a flat scan under the same graded score. On EPBench the index cuts the candidate set from 200 to 19.5 cells on average, and the share of questions whose top 8 contains a gold event moves from 0.983 to 0.975. On LoCoMo the candidate set shrinks from 601.7 to 351.2 turns, with recall 0.907 versus 0.906.

At this memory size run time is small. Scoring only the candidates takes 0.047 ms per query, which is 2.9× faster than the flat scan (0.135 ms), and both are small next to the 7.7 ms needed to embed a new cue string. Appendix C gives the masked timing and a synthetic scale-up, and withdraws an earlier 7.7-fold speed-up.

6 Discussion

The clearest result is that a hard intersection fails when an LLM extracts the cues, and that almost any rule that adds up the per-axis matches repairs most of the damage. It held on both books and under a relaxed threshold. In practice, a hard threshold on LLM-extracted cues is a risky default.

The second result concerns how many events to hand to the answering model. A ranked set of K events suits a question that asks for a list, but it hurts a question that asks for one extreme, such as the latest event, because later events that are not the answer slip in. The rule should depend on the kind of question. This rests on 48 "latest" questions from one benchmark. A router that sends them to the stricter rule recovers the loss without a judge, although it reads the question type from the benchmark label.

Third, confidence is informative, since wrong cues get lower confidence, but as a weight it changes almost nothing. As a gate it helps a little (Gated-AND beats Hard-AND by 0.066 F1) but still returns nothing for 38.9% of the answerable questions. One gap remains. The soft rules cannot say "nothing found": the graded sum does so correctly for only 0.6% of the 180 questions with an empty answer (Table 1). A practical memory therefore needs a way to abstain.

7 Conclusion

We asked what keeps structured retrieval working when an LLM extracts the cues. Do not require every cue to match: a hard intersection left 63% of the EPBench questions without context, and adding up the per-axis matches recovered most of that. Choose the rule by what the question asks: a ranked set helps when it lists events and hurts when it asks for the latest one. A cue's confidence tracks whether the cue is right, but as a weight it changed neither retrieval nor accuracy. On LoCoMo, matching the speaker raised answer accuracy over dense retrieval, and the dense term stayed necessary. We claim nothing beyond these benchmarks.

Limitations

1. We tested cue confidence as a single weight per axis and as a gate (Gated-AND); other uses, such as a threshold or context size that depends on the confidence, are left for future work.
2. The dense baseline is one retriever (BGE-base). Hard-AND and Gated-AND are our reimplementation of the published rule, and we did not compare with other memory systems or make claims relative to them.
3. The judgments come from LLM judges. On LoCoMo the answerer and the judge are the same model, so another model re-graded three of the seven variants (Section 5.4); we did not check the judges against humans, and the EPBench judgments were not re-graded.
4. Intervals reflect question sampling, with a conversation-level check on LoCoMo; EPBench has two memories, and each system ran once.
5. The evidence comes from two synthetic EPBench books and one dialogue benchmark, so we limit our claims to these settings.

Ethical Considerations

Data. EPBench is synthetic, and LoCoMo consists of conversations between fictional personas that were generated by language models and edited by human annotators. We chose them so that no personal data are involved, and we collected none. Such data only approximate real use. Our longer-term aim for this line of work is to test the memory on data that represent real use accurately, which for a personal memory means real people's conversations and routines. We would do this only with informed consent, anonymization, approval from an ethics board, and clear retention and deletion rules, and we would report those results separately from the synthetic ones.

Deployment. A deployed memory raises further risks that our benchmarks do not test. A memory that always returns something can present irrelevant content with unwarranted authority, and our results on empty-gold questions quantify this. Confidence errors can surface sensitive or private memories, or hide relevant ones. A cue can be matched to the wrong person when several personas share a conversation. Old information can be retrieved as if it were current, and long retention of personal memories needs deletion and consent policies. Any deployment should add abstention, access control and retention limits. Our results on synthetic data should not be read as evidence of safety in such settings.

References

Poo-Hee Chang and Ah-Hwee Tan. 2017. Encoding and recall of spatio-temporal episodic memory in real time. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI-17)*, pages 1490–1496.

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

DeepSeek-AI. 2025. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *Preprint*, arXiv:2501.12948.

Bradley Efron and Robert J. Tibshirani. 1993. *An Introduction to the Bootstrap*. Chapman & Hall.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1321–1330.

Bernal Jiménez Gutiérrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. 2024. HippoRAG: Neurobiologically inspired long-term memory for large language models. In *Advances in Neural Information Processing Systems*, volume 37, pages 59532–59569.

Bernal Jiménez Gutiérrez, Yiheng Shu, Weijian Qi, Sizhe Zhou, and Yu Su. 2025. From RAG to memory: Non-parametric continual learning for large language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 21497–21515.

Chuanrui Hu, Xingze Gao, Zuyi Zhou, Dannong Xu, Yi Bai, Xintong Li, Hui Zhang, Tong Li, Chong Zhang, Lidong Bing, and Yafeng Deng. 2026. EverMemOS: A self-organizing memory operating system for structured long-horizon reasoning. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 45836–45853, San Diego, California, United States. Association for Computational Linguistics.

Alexis Huet, Zied Ben Houidi, and Dario Rossi. 2025. Episodic memories generation and evaluation benchmark for large language models. In *The Thirteenth International Conference on Learning Representations (ICLR)*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474.

Zhiyu Li, Chenyang Xi, Chunyu Li, Ding Chen, Boyu Chen, Shichao Song, et al. 2025. MemOS: A memory OS for AI system. *Preprint*, arXiv:2507.03724.

Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. 2024. Evaluating very long-term conversational memory of LLM agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13851–13870, Bangkok, Thailand. Association for Computational Linguistics.

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST)*, pages 1–22.

Qwen Team. 2024. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Preston Rasmussen, Pavlo Paliychuk, Travis Beauvais, Jack Ryan, and Daniel Chalef. 2025. Zep: A temporal knowledge graph architecture for agent memory. *Preprint*, arXiv:2501.13956.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 3982–3992.

Ah-Hwee Tan, Budhitama Subagdja, Di Wang, and Lei Meng. 2019. Self-organizing neural networks for universal learning and multimodal memory encoding. *Neural Networks*, 120:58–73.

Cassandra Hui-Ming Tan, Budhitama Subagdja, and Ah-Hwee Tan. 2026. ARTEM: Enhancing large language model agents with spatial-temporal episodic memory. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 25753–25760.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D. Manning. 2023. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5433–5442.

Endel Tulving. 1983. *Elements of Episodic Memory*. Oxford University Press.

Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. 2023. C-Pack: Packed resources for general Chinese embeddings. *Preprint*, arXiv:2309.07597.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2024. Can LLMs express their uncertainty? an empirical evaluation of confidence elicitation in LLMs. In *The Twelfth International Conference on Learning Representations (ICLR)*.

Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. 2025. A-MEM: Agentic memory for LLM agents. *Preprint*, arXiv:2502.12110.

Bianca Zadrozny and Charles Elkan. 2002. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 694–699.

A Notation

Q: query. $A(Q)$: axes constrained by Q. $S(e)$: score of cell e. $s_{\text{den}}, s_{\text{per}}, s_{\text{top}}$: dense, persona and topic terms of the LoCoMo score, with weights w_p and w_t. C_k: extracted cue. $\hat{c}_k$: self-reported confidence. g_k: isotonic calibrator. $\text{Conf}_k = g_k(\hat{c}_k)$. $m_k \in [0, 1]$: graded match. ρ_k: vigilance threshold. γ: gating threshold. "Top-K" is the context size: 8 events on EPBench and 50 turns on LoCoMo.

B Reproducibility details

Models. EPBench events: Qwen2.5-32B-Instruct. EPBench cues, answers and judgments: DeepSeek-R1-Distill-Qwen-14B, deterministic decoding; 512 new tokens for cues. LoCoMo topic and persona extraction: Qwen2.5-7B-Instruct. LoCoMo answers and judgments: Qwen2.5-32B-Instruct. Embeddings: all-MiniLM-L6-v2 (EPBench) and BAAI/bge-base-en-v1.5 (LoCoMo). All models ran for inference only, on one A100 80GB GPU. The seven end-to-end LoCoMo runs of Table 3 used vLLM (bf16, greedy decoding, at most 700 new tokens for answers and 512 for judgments, FlashInfer sampler switched off because its kernel failed to compile; we did not record the vLLM version). The earlier confidence runs of Table 10 used Hugging Face generate. On the same 1,536 questions the new engine scores -2.1 points ($[-4.6, 0.5]$) against the earlier binary run, and -2.0 ($[-3.7, -0.2]$) and -2.3

($[-4.0, -0.5]$) for the full graded score against the earlier no-confidence and raw-confidence runs, whose retrieved sets are almost the same.

Independent judge. claude-sonnet-5 was called through the platform's language-model interface with the Qwen judge prompt and system message (below) and a limit of 150 new tokens for the first 1,156 calls and 1,500 for the rest, because the short limit returned empty replies for part of the calls, which were re-run. Sampling settings are the platform's defaults and were not recorded. A per-session token ceiling stopped the calls after 2,340 of the 6,817 distinct (question, gold, prediction) triples. These cover all answers of dense, full and held-out weights on questions 0 to 1,035, and parts of the other variants, which we do not use. Four judgments carried no label, so 1,032 questions are complete.

EPBench answer and judge prompts. We use the ARTEM evaluation code and prompts unchanged. We only raised the judge's generation limit from 4,096 to 8,192 tokens, to reduce truncated judgments, and added per-question checkpointing.

Match functions. EPBench time: $1 - |t_Q - t_e|$ on dates scaled to $[0, 1]$. EPBench place, entity and content: $\sum_i \min(q_i, w_i)/384$ over the complement-coded, per-vector min–max scaled MiniLM embeddings of the cue and of the cell field, as in STEM. LoCoMo dense term: cosine similarity of BGE embeddings, with the query prefix "Represent this sentence for searching relevant passages: ". LoCoMo topic term: cosine similarity between the BGE embedding of the stored anchor topic and that of the question topic.

Ties and decoding. The graded rule keeps the K highest scores with NumPy's default `argsort`, which is not stable. Equal scores are therefore ordered arbitrarily but deterministically, which explains the 3 of 668 differences in the reproduction check below. Hard-AND, Gated-AND and Literal return sets and have no tie ordering. MiniLM vectors for the EPBench match are not length-normalised: they are scaled per vector to $[0, 1]$ and complement-coded. BGE vectors are L2-normalised. All LLM calls use deterministic decoding, and the bootstrap seed is 0.

Calibration protocol. EPBench: reference cues from the benchmark. Five-fold cross-fitting scores 546 questions, and a calibrator fitted on those scores the 122 held-out questions. LoCoMo: leave-one-conversation-out, against the gold speaker for persona and against the gold evidence fact (cosine above a fixed threshold) for topic. Isotonic regression uses scikit-learn defaults with values clipped to $[0, 1]$.

$w_p \backslash w_t$	0.00	0.10	0.15	0.20	0.30	0.40
0.00	.774	.785	.792	.795	.792	.786
0.10	.885	.893	.895	.895	.881	.876
0.15	.891	.897	.898	.895	.889	.884
0.20	.896	.902	**.904**	.901	.893	.884
0.30	.887	.896	.897	.894	.886	.880
0.40	.885	.893	.895	.891	.882	.875

Table 4: Weight grid for the LoCoMo score (Eq. 1). Values are the share of gold evidence turns that reach the answering model, over 1,536 questions.

Weights. Table 4 lists the grid used to choose the LoCoMo weights, on the metric "evidence to LLM". We chose the best cell, $w_p = 0.2$ and $w_t = 0.15$. The same question pool is used for the evaluation. The grid uses constant confidences (0.95 for the speaker, 0.70 for the topic), as in the original sweep script. Table 9 uses the stored calibrated confidences instead, and our offline re-run of the grid reproduces the logged values (0.774 for dense retrieval, 0.904 at the chosen cell).

Cue-extraction prompt (EPBench). Braces are shown as in the rendered prompt, and an arrow is written in ASCII.

```
You extract SEARCH CONSTRAINTS from an episodic-memory
      question.

A question asks to find events filtered by some of these axes:
- time     : a date the events happened (e.g. "April 09,
      2226")
- spaces   : a location (e.g. "Mars Valles Industrial Hub")
- entities : a person/protagonist name (e.g. "Aurora Chavez")
- content  : an event type/topic (e.g. "magnetic confinement
      rupture")

Extract ONLY the GIVEN constraints in the QUESTION, NOT the
      thing being asked for.
Example: "list all DATES of events about X" -> the date is the
       ANSWER, set time=null;
only X (content) is the constraint. For each axis give the
      value and your confidence
0.0-1.0 that you extracted it correctly. If an axis is not a
      given constraint, output null.

Return ONLY a JSON object, no other text:
{"time": {"value":"...","confidence":0.0} or null,
 "spaces": {"value":"...","confidence":0.0} or null,
 "entities": {"value":"...","confidence":0.0} or null,
 "content": {"value":"...","confidence":0.0} or null}

QUESTION: "{question}"
```

Anchor-extraction prompt (LoCoMo). System message: "You are a precise information extractor.

Output JSON only, no prose."

```
Extract SEMANTIC ANCHORS from one conversation session (
      speakers: {spk}).

For each MEANINGFUL fact/event in the session, output:
  - "persona": the person it is ABOUT (a speaker name or a
      named third person),
  - "persona_confidence": your own confidence (0.0-1.0) that "
      persona" is correct -
                1.0 if the person is explicitly named/
      unambiguous, lower if inferred/guessed,
  - "topic"  : a SHORT canonical phrase (2-4 words: an event
      type / subject),
                NOT a full sentence, NO dates, NO locations,
  - "topic_confidence": your own confidence (0.0-1.0) that "
      topic" faithfully summarizes
                the fact - lower if the turn is vague,
      sarcastic, or you had to guess intent,
  - "dia_id" : the id of the turn where that fact is primarily
       stated (e.g. "D2:8").

Ignore greetings, small-talk, and pure reactions ("that's cool
      !"). One turn may yield
0, 1 or several anchors. Use ONLY dia_ids that appear below.
      Do NOT default every
confidence to 1.0 - vary them honestly based on how certain
      you actually are.

Return a SINGLE JSON dict:
{"anchors": [{"persona":"Caroline","persona_confidence":0.95,"
      topic":"adoption research","topic_confidence":0.8,"
      dia_id":"D2:8"}]}

Session turns:
{turns}
```

LoCoMo answer prompt. System message: "Reason step by step, then give FINAL ANSWER." The opening sentence is adapted from EverMemOS.

```
You are an intelligent memory assistant answering a question
      from conversation memories between two speakers.
      Memories are timestamped.

Follow this Chain-of-Thought, then give a concise final answer:

STEP 1 - list the memories (with dates) relevant to the
      question.
STEP 2 - extract exact names, numbers, dates, frequencies.
STEP 3 - link entities across memories and make reasonable
      INFERENCES (don't dismiss as speculation).
STEP 4 - TIME CALCULATION: convert relative time to absolute
      using the memory dates (e.g. "last year" from a May 2023
       message -> 2022; "yesterday" of an 8 May message -> 7
      May).
STEP 5 - answer.

Rules: preserve exact names/numbers/dates; if truly
      unsupported, say "no information".
End with a line exactly: FINAL ANSWER: <concise answer>

{context}

Question: {question}
```

LoCoMo judge prompt. System message: "You are an expert grader that determines if answers match a gold standard answer."

```
Your task is to label an answer to a question as CORRECT or
      WRONG. You are given:
(1) a question (posed by one user to another user),
(2) a 'gold' (ground truth) answer,
(3) a generated answer.

The point of the question is to ask about something one user
      should know about the other based on their prior
      conversations. The gold answer is usually concise and
      short. The generated answer might be much longer, but
      you should BE GENEROUS with your grading - as long as it
       TOUCHES ON THE SAME TOPIC as the gold answer, it should
       be counted as CORRECT.

For time-related questions, the gold answer is a specific date
      /month/year. The generated answer might be longer or use
       relative references (like 'last Tuesday' or 'next month
      '), but be generous - as long as it refers to the SAME
      date or time period as the gold answer, count it CORRECT.
       Even if the format differs (e.g., 'May 7th' vs '7 May'),
       consider it CORRECT if it's the same date.

Question: {q}
Gold answer: {gold}
Generated answer: {pred}

First provide a short (one sentence) explanation, then finish
      with CORRECT or WRONG.
```

Offline re-runs. The confidence and K analyses re-run the retrieval code offline with MiniLM embeddings and no LLM. This reproduces the stored retrieved events for 668 of 668 questions under Hard-AND, Gated-AND and Literal. It does so for 665 of 668 under Graded, where the three differences are tie orderings.

Earlier exploratory variants (not re-run). On an evidence-recall metric over LoCoMo, BM25 plus dense fusion (0.757), an agentic sufficiency loop (0.710, n=100) and a cross-encoder reranker (0.783) all fell below the dense baseline (0.774) or the graded score (0.904). Extending the persona axis from the speaker to any mention lowered the score from 0.904 to 0.881.

C Additional tables

Run time and scale (Section 5.5). The repository's index-first code scores all cells and then masks the non-candidates, so the candidate count alone saves no compute. After warm-up, the flat scan takes 0.135 ms per query and the masked version 0.140 ms. Scoring only the candidates takes 0.047 ms, which is 2.9× faster than the flat scan. All three are small next to the 7.7 ms needed to embed a new cue string. An earlier 7.7-fold speed-up is withdrawn: it ran the flat scan first, which paid for query encoding alone. A synthetic scale-up with random vectors (Table 13, Appendix C) measures compute only. The flat scan takes 25 ms per query at 20,000 cells and 65 ms at 50,000, and scoring only the candidates is 8 to 13 times faster at the measured 9.75% share and 174 to 195 times faster at 1%. We did not measure the candidate share or retrieval quality of a real memory that large.

Rule	Empty ctx. (%)	Empty, answ. (%)	F1 lenient [95% CI]	F1 harsh [95% CI]	Ctx. recall
Book 1 (Uscifi, 668 questions, 611 judge-valid)					
Hard-AND	63.2	49.6	0.508 [0.470, 0.545]	0.486 [0.449, 0.524]	0.438
Gated-AND	55.2	38.9	0.574 [0.537, 0.610]	0.550 [0.514, 0.587]	0.526
Literal	31.3	14.5	0.725 [0.692, 0.757]	0.685 [0.653, 0.718]	0.745
CIRCUIT	1.0	1.2	0.748 [0.718, 0.778]	0.713 [0.682, 0.744]	0.808
Book 2 (Udefault-196, 686 questions, 634 judge-valid)					
Hard-AND	59.3	45.3	0.548 [0.512, 0.586]	0.527 [0.491, 0.563]	–
Literal	25.7	13.8	0.716 [0.684, 0.748]	0.684 [0.652, 0.715]	–
CIRCUIT	2.2	2.2	0.735 [0.705, 0.765]	0.705 [0.674, 0.734]	–

Table 5: EPBench with LLM-extracted events and cues, identical candidates and judge across rules. "Empty ctx." is the share of all questions for which nothing is retrieved; "answ." restricts to questions with a non-empty gold answer (Book 1: 488, Book 2: 506). F1 and context recall are computed on the questions whose judge output parsed for every rule. Context recall is the share of gold answer items found verbatim in the retrieved events (Book 1, answerable questions).

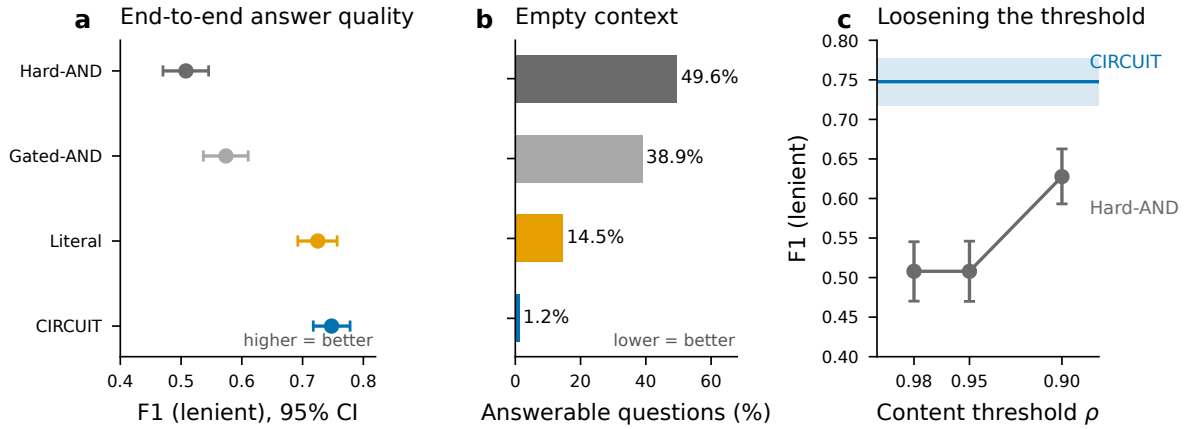


Figure 3: EPBench Book 1. (a) Lenient F1 on 611 judge-valid questions. (b) Share of the 488 answerable questions with an empty context. (c) Hard-AND with a relaxed content threshold ρ; the band is the graded sum. All intervals are 95% bootstrap intervals. The $\rho = 0.85$ run was not judged.

Rule	Empty (%)	List or order	Latest
Dense, top 8 events	0.0	0.518	0.042
Dense, top 20 events	0.0	0.645	0.083
$\sum_k m_k$, top 8	1.0	0.920	0.292
$\max_k m_k$, top 8	1.0	0.845	0.292
$\min_k m_k$, top 8	3.9	0.858	0.125
$\prod_k m_k$, top 8	3.9	0.913	0.188
Graded, $\sum_k \mathrm{Conf}_k m_k$, top 8	1.0	0.916	0.312
Graded score ≥ 0.90	19.3	0.854	0.667
Graded score ≥ 0.95	46.0	0.647	0.417
Literal	31.3	0.800	0.792

Table 6: Soft aggregation rules and dense retrieval on EPBench Book 1, without an LLM judge. Empty: share of all 668 questions with an empty context. List or order and Latest: gold-chapter recall on the 440 and 48 answerable questions of those types. Dense: MiniLM similarity between the question and the event text. Calibrated confidence in the graded rules.

Rule	Cue confidence	Empty (%)	Gold-chapter recall	Sets differ
Literal	calibrated	31.3	0.799 [0.766, 0.830]	1
Literal	raw	31.1	0.801 [0.769, 0.833]	0
Literal	constant (1.0)	31.1	0.801 [0.768, 0.833]	–
Graded	calibrated	1.0	0.857 [0.831, 0.882]	41
Graded	raw	1.0	0.859 [0.832, 0.884]	5
Graded	constant (1.0)	1.0	0.858 [0.832, 0.883]	–

Table 7: Cue-confidence ablation on EPBench Book 1 without an LLM judge. Recall is over the 488 answerable questions. "Sets differ" counts questions (of 668) whose retrieved events differ from the constant-confidence run.

Axis	Right	Wrong	Raw conf. (right / wrong)	Calibrated
Time	205	19	1.00 / 0.88	0.99 / 0.14
Place	219	0	0.99 / –	0.99 / –
Entity	327	0	0.98 / –	0.98 / –
Content	242	48	0.98 / 0.91	0.89 / 0.58

Table 8: Mean cue confidence for extracted cues that agree (right) or disagree (wrong) with the reference cue, EPBench Book 1.

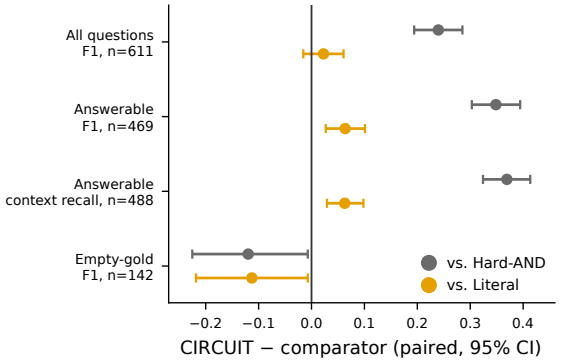


Figure 4: Paired differences, graded sum (CIRCUIT) minus comparator, on EPBench Book 1 with 95% bootstrap intervals. Positive favors the graded sum. F1 is lenient and uses judge-valid questions. "Answerable" means a non-empty gold answer.

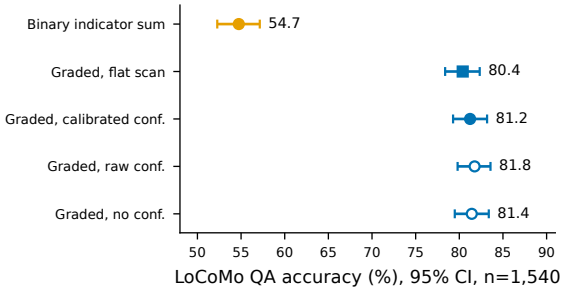


Figure 5: LoCoMo accuracy of the earlier runs (Hugging Face engine) with 95% bootstrap intervals ($n = 1{,}540$). Square: no index (flat scan). Filled circle: calibrated confidence. Open circles: raw or no confidence. Orange: binary topic term without the dense term.

LoCoMo retrieval	Any gold (%)	All gold (%)	Any, minus dense [95% CI]
Dense only	77.4	64.0	–
+ speaker	89.5	78.4	+12.0 [10.2, 13.9]
+ topic only	78.8	65.0	+1.4 [0.2, 2.5]
+ both, calibrated	89.6	78.3	+12.2 [10.2, 14.1]
+ both, raw	89.6	78.3	+12.2 [10.2, 14.1]
+ both, no confidence	89.5	77.8	+12.1 [10.1, 14.1]
Binary, no dense term	54.1	40.8	-23.3 [-26.2, -20.4]

Table 9: LoCoMo retrieval without an LLM, on the 1,536 questions with gold turn ids. Any gold (all gold): share of questions whose context holds a gold turn (all gold turns). Context: top 10 sessions, 50 turns; BGE-base; w_p=0.2, w_t=0.15 (the flat-scan accuracy run in Table 10 used w_p=0.15; the any-gold share is then 89.3%). The dense baseline has 77.4% [75.3, 79.5]. The last column is the paired difference to dense retrieval in points.

Variant	Tokens	Single (841)	Multi (282)	Temp. (321)	Open (96)	Overall [95% CI]
Graded, flat scan (no index)	1590	84.5	78.7	79.4	52.1	80.4 [78.4, 82.3]
Graded, two-tier, calibrated conf.	1676	85.1	79.8	79.4	57.3	81.2 [79.3, 83.2]
Graded, two-tier, raw conf.	1755	85.5	81.2	79.4	58.3	81.8 [79.8, 83.6]
Graded, two-tier, no conf.	1773	84.7	80.1	81.0	58.3	81.4 [79.5, 83.4]
Binary indicator sum, two-tier	1483	53.2	66.7	52.0	42.7	54.7 [52.3, 57.1]

Table 10: Earlier runs (Hugging Face generate; Appendix B). LoCoMo QA accuracy (%) on the 1,540 answerable questions of all ten conversations (Qwen2.5-32B-Instruct answerer and judge). Category sizes in parentheses. The four two-tier variants share candidates, cues, weights (w_p=0.2, w_t=0.15) and judge; only the combination rule or the confidence source changes. The flat scan scores all turns and used w_p=0.15.

	Qwen2.5-32B judge	Independent judge
Dense only	74.0 [71.3, 76.5]	67.9 [65.0, 70.7]
Dense + speaker + topic (full)	79.1 [76.7, 81.5]	75.0 [72.4, 77.6]
Full, held-out weights	80.7 [78.3, 83.0]	76.1 [73.5, 78.6]
Full minus dense	+5.0 [+2.6, +7.6]	+7.1 [+4.5, +9.6]
Held-out minus full	+1.6 [-0.1, +3.4]	+1.1 [-0.7, +2.8]
Held-out minus dense	+6.7 [+4.4, +9.0]	+8.1 [+5.6, +10.8]

Table 11: Independent-judge check on 1,032 LoCoMo questions of seven conversations (3,096 answers of the three variants). Accuracy (%) and paired differences in points with 95% bootstrap intervals over questions. The two judges agree on 92.9% of the answers ($\kappa = 0.81$). Qwen marks 186 answers correct that the independent judge marks wrong, and 33 the other way round. The independent judge is claude-sonnet-5, with the judge prompt of Appendix B.

	EPBench	LoCoMo
Cells in memory	200	about 600
Candidate set (mean)	19.5	351.2 of 601.7
Reduction	-90.3%	-41.6%
Gold event in top-K, flat	0.983	0.907
Gold event in top-K, index-first	0.975	0.906
Scoring time per query, flat (ms)	0.135	–
Index-first, scores all cells then masks	0.140	–
Index-first, scores candidates only	0.047	–
Embedding one new cue string (ms)	7.7	–
Build index maps and cells (ms)	3.0	–
Embed the events once (s)	2.9	–
Index maps in memory (MB)	0.58	–

Table 12: Index-first versus flat-scan retrieval under the same graded score (K=8 on EPBench, K=50 on LoCoMo). "Gold event in top-K" is the share of answerable questions whose context holds a gold cell. Times are medians of 7 warm repetitions over 661 queries on one CPU thread. The LoCoMo column is from earlier single-run measurements and was not repeated.

N	Candidates (%)	Candidates	Flat (ms)	Masked (ms)	Candidates only (ms)	Speed-up	Index build (s)	Index (MB)	Vectors (MB)
200	9.75	20	0.11	0.112	0.0245	4.5	0.0016	0.1	1
200	1.00	10	0.11	0.112	0.0172	6.4	0.0016	0.1	1
2,000	9.75	195	1.31	1.32	0.143	9.2	0.014	1.5	12
2,000	1.00	20	1.31	1.32	0.0239	54.8	0.014	1.5	12
20,000	9.75	1,950	25	25.3	1.92	13.0	0.15	13.1	117
20,000	1.00	200	25	25.3	0.144	173.8	0.15	13.1	117
50,000	9.75	4,875	64.8	65.1	7.77	8.3	0.37	35.6	293
50,000	1.00	500	64.8	65.1	0.333	194.5	0.37	35.6	293

Table 13: Synthetic scale-up of the EPBench retrieval cost (one CPU thread; for each timing, the median of 5–15 repetitions of 30 queries, and the fastest of three passes). Cells carry random 768-dimensional complement-coded vectors on two axes, as in the real engine; the candidate set is a random subset with the given share. 9.75% is the mean share measured on EPBench at $N = 200$. Flat and masked times do not depend on the candidate share, so each size reports the fastest of its three measurements (one first measurement at $N = 50{,}000$ was 94 ms). Speed-up is flat time over candidates-only time; the masked variant scores all cells and costs the same as the flat scan. Index size and build time come from synthetic token maps. The table measures compute only: it does not measure the candidate share or the retrieval quality of a real large memory.

Axis	Cues	Correct	ECE raw	ECE calibrated
Book 1 (five-fold cross-fit)				
Time	224	205/224 (0.92)	0.074	0.013
Place	219	219/219 (1.00)	0.007	0.007
Entity	327	327/327 (1.00)	0.019	0.019
Content	290	242/290 (0.83)	0.135	0.069
Book 2 (Book-1 calibrator, unchanged)				
Time	217	202/217 (0.93)	0.061	0.005
Place	224	224/224 (1.00)	0.007	0.007
Entity	345	344/345 (1.00)	0.016	0.016
Content	258	209/258 (0.81)	0.158	0.065

Table 14: Cue-confidence calibration on EPBench. ECE uses 10 equal-width bins. Book 1 confidences come from five-fold cross-fitting, so no question is scored by a calibrator fitted on its own label. Book 2 uses the Book-1 calibrator without refitting. Place and entity cues are almost always correct, so there is little label variance to fit.

ρ_{content}	Empty ctx.	Hard-AND F1	CIRCUIT − Hard-AND
0.98	422/668 (63.2)	0.508 [0.470, 0.545]	0.240 [0.194, 0.285]
0.95	422/668 (63.2)	0.508 [0.470, 0.546]	0.240 [0.195, 0.284]
0.90	320/668 (47.9)	0.628 [0.593, 0.663]	0.120 [0.079, 0.161]
0.85	257/668 (38.5)	not judged	–

Table 15: Sensitivity of Hard-AND to the content vigilance threshold (other axes fixed at the original values; Book 1, 611 judge-valid questions). Empty-context counts are exact retrieval-only counts over all 668 questions. The $\rho = 0.85$ run was retrieved but not judged.

Rule	Judge failures	F1 len.	F1 harsh	Empty ctx.
Hard-AND	17	0.504	0.497	422/668
Gated-AND	19	0.568	0.562	369/668
Literal	23	0.702	0.689	209/668
CIRCUIT	33	0.706	0.709	7/668

Table 16: EPBench Book 1 without excluding judge failures. Lenient F1 scores a failed judge call as 0 over all 668 questions; harsh F1 averages only over parsed calls.

Question asks for	n	Hard-AND	Gated-AND	Literal	CIRCUIT
Entities	123	0.45 / 70	0.47 / 68	0.69 / 35	0.77 / 0
Event contents	149	0.66 / 41	0.74 / 32	0.75 / 22	0.69 / 0
Spaces	156	0.54 / 65	0.58 / 58	0.74 / 35	0.75 / 3
Times	166	0.42 / 72	0.53 / 59	0.74 / 35	0.81 / 1

Table 17: EPBench Book 1, by retrieval type: lenient F1 on judge-valid questions / percentage of questions of that type with an empty context (all questions of the type). Types with fewer than 20 valid questions are omitted.