Tracing False Speaker Links Through Retrieval and Answering

Dev D. Goyal
Independent Researcher
dvgyl@outlook.com
ORCID: 0009-0009-9132-744X

Abstract

Memory systems for large language model (LLM) agents can organize past conversations by person, for example in a speaker index or a knowledge graph. To answer "Who promised the budget?", such a system searches the relevant people's messages and passes a few to the LLM. Building the index requires entity resolution, which can wrongly link two people. A false link can change what the LLM reads and make it name the wrong person, but final-answer scores may not show this. We add false links at known rates to a retrieval pipeline, keep every message and name unchanged, and follow 280 questions from two synthetic benchmarks from search to answer. With one false link per 40 people, almost one in four workplace searches reaches a wrong person. At the highest rate, gpt-oss-120b reads a different context in 81.0% of workplace runs, but its accuracy falls by 1.75 percentage points, because improved and worsened answers offset each other. For gpt-oss on the social benchmark, one question that loses its only labeled supporting message accounts for 79% of history-weighted loss. Accuracy alone can therefore hide identity errors. We provide a tool and a memory error card that measure each step.

1 Introduction

Consider a board-game group chat from the SocialMemBench benchmark (Owolabi, 2026). Two members say that players should read the rules before game night. Tomasz says it in general terms. Marcus is specific: "whoever owns it should send the rulebook link by Wednesday." A user asks a memory assistant which member gave the specific procedure. With a correct speaker index, the gpt-oss-120b model answers Marcus in all four runs. We then add five false links to the index. One of them wrongly joins Marcus with Derek, so the search also collects Derek's messages, such as "agree. read the rules people." The model can read only a limited amount of text. The changed selection omits Marcus's labeled statement, and the model answers Tomasz instead. No name and no message changed; only the selection did (Figure 1).

Assistants with long-term memory store past conversations and, for each question, retrieve a few relevant messages for a large language model (LLM) to read (Packer et al., 2023; Chhikara et al., 2025). Many memory systems organize what they store around people and other entities, for example in an entity graph (Rasmussen et al., 2025; Gutiérrez et al., 2025). We study a simple form of this idea: a *speaker index* that groups messages by the person who wrote them, so that the search can focus on the people a question is about. To build such an index, a system must decide which names refer to the same person. This task, called *entity resolution*, makes mistakes: it can join two different people, and one wrong join can lead to others (Baas et al., 2021; Pavani et al., 2026). SocialMemBench reports such identity problems in memory for groups. We call a wrong join a *false link*.

These errors matter for two reasons. First, they can reach users. A false link can make the assistant credit the wrong person, as in the Marcus example, although every stored message is correct, and it adds messages from people the question is not about to the model's input. Second, a score based only on final answers may not show them. A changed search can return the same text, a changed text can produce the same answer, and when answers change, gains can cancel losses. Studies of model updates and compression call such changes churn or flips (Milani Fard et al., 2016; Dutta et al., 2024). A memory system can therefore look robust to identity errors while some of its answers fail.

We measure where the effect of a false link goes.

Proceedings of the 3rd Workshop on KNOwledge GRaphs & Agentic Systems Interplay (NORA 2026), pages 109–130
November 10, 2026 ©2026 Association for Computational Linguistics

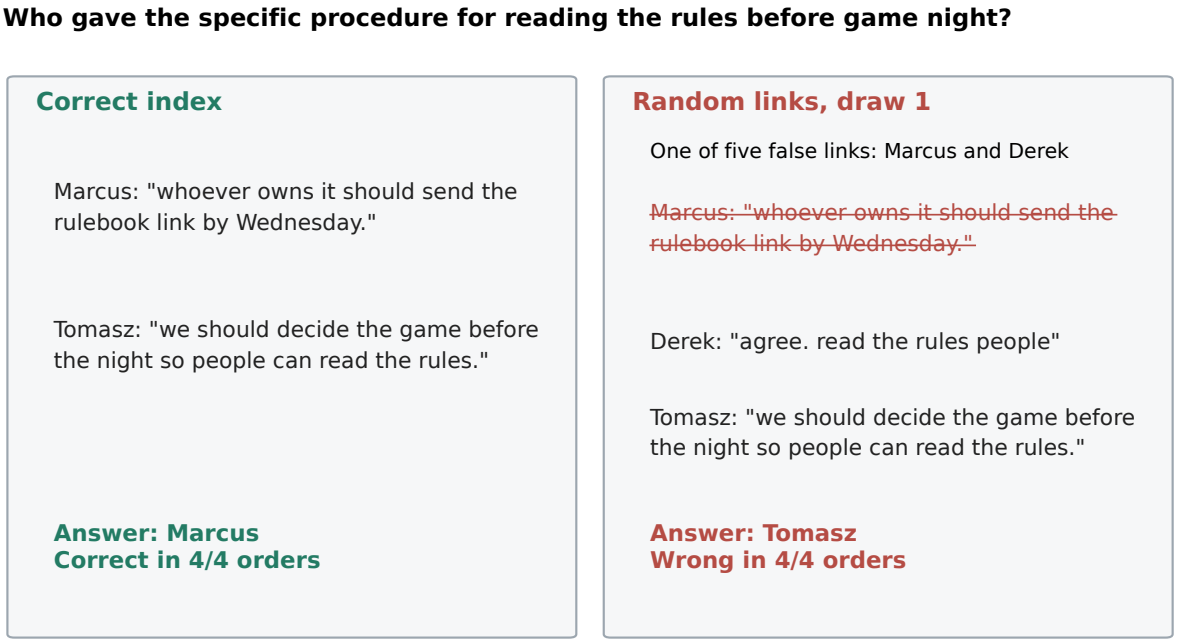


Figure 1: With false links, the only labeled supporting message is absent and the model names the wrong person, although every name stays correct. The panels show selected messages for the Marcus question. The first random set of five false links includes one between Marcus and Derek. Two Derek messages and one Soo-Yeon message enter; Marcus's labeled statement and one Priya message leave. Over three random sets and four option orders, gpt-oss selects Tomasz in 10 of 12 runs. We selected this case after decomposing the losses. Appendix H gives the complete trace and the contrasting Qwen result.

We add false links at known rates to the speaker index of a retrieval pipeline and keep every message and name unchanged. We answer each question with the correct index and with the false links, and we record five steps for both: the people that the search reaches, the shortlisted messages, the text that the model reads, its answer, and whether the answer is correct. We call this paired record a *stage audit*. We test 280 questions from two synthetic benchmarks, one about workplace projects and one about social groups, with three LLMs. Unlike evaluations that compare only final scores, this design keeps every message and name fixed and shows the step at which each change stops.

We find three patterns. First, false links reach many questions even when they are rare, because each search starts from several people: with one false link per 40 people, 23.6% of workplace searches reach a wrong person (Section 4.1). Second, accuracy hides most of the change: at the highest rate, gpt-oss reads a different text in 81.0% of workplace runs and changes 19.25% of its answers, but its accuracy falls by only 1.75 percentage points (Section 4.2). Third, gpt-oss losses concentrate in a few social questions: 5 of 80 questions carry all lost answers, and one carries 79% of history-weighted loss (Section 4.3).

Our contributions are:

1. A controlled test of speaker-index errors that changes only which messages the search may use, while every message and name stays correct.
2. Evidence that final accuracy hides both widespread answer changes and concentrated failures, with a simple formula for the share of questions that an error rate reaches.
3. An offline tool and a one-page memory error card that let other teams measure each step of their own memory systems, including loss and gain rates that account for different starting accuracy.[1]

2 System and intervention

2.1 From a question to a context

Retrieval finds past messages that may help answer a question. Our pipeline combines meaning-based search with BGE-small-en-v1.5 and word-based search with BM25 (Xiao et al., 2024;

[1]Code and memory error card: https://github.com/dvgyl/false-speaker-links

Robertson and Zaragoza, 2009). Reciprocal rank fusion combines their rankings, and the top eight messages provide the starting speakers (Cormack et al., 2009).

Under the *correct index*, each connected group contains one true person. The speaker filter admits messages from starting speakers and anyone linked to them, including through a chain. The first 64 eligible messages form the *shortlist*, where a *reranker* scores each question–message pair for relevance (Sentence Transformers, n.d.).

The pipeline scans the shortlist in score order and keeps each whole message that fits the remaining *context budget* of 2,048 tokens, the text units counted by the tokenizer. Selected messages form the *context*, which goes in time order to the *reader*, the model that answers the question. Appendix A gives scoring and serialization details.

False links preserve source text, names, relevance scores, and message costs. *Unfiltered search* removes the speaker filter and takes the 64 highest-ranked messages from the whole history. False links relax the filter, so they can also restore useful messages that the correct index excludes. The intervention can therefore produce gains as well as losses.

2.2 False links and query exposure

For a history with $n \geq 2$ speakers, random links connect each distinct pair independently with probability $d/(n-1)$. The *link rate* d is the expected number of incident false links per person. Thus $d = .05$ means one undirected false edge per 40 people on average. At this small rate, about one person in 20 has an incident false link. We test $d \in \{.05, .2, .8\}$ with ten fixed *draws*, each a sampled link set. Within a draw, higher rates add links to the same set, and we retain draws with no links. *Look-alike links* favor similar names while matching each random draw's edge count (Appendix B).

A search starts from several people, so a sparse graph can expose many questions to a false link. With K starting speakers, the chance of reaching at least one new person is

$$P(\text{new person}) = 1 - \left(1 - \frac{d}{n-1}\right)^{K(n-K)}. \quad (1)$$

For small pair probabilities and $K \ll n$, this is approximately $1 - e^{-Kd}$. For example, $K = 5$ and $d = .05$ give about 22% exposure. Exposure is the first opportunity for a retrieval change, not an answer-error probability.

3 Experimental design

3.1 Workloads, readers, and controls

EverMemBench-Dynamic supplies five workplace conversation histories (Hu et al., 2026). Fixed hash selection chooses 40 four-option questions per history, preserving six eligible question families. SocialMemBench supplies all 80 attribution questions from 41 histories (Owolabi, 2026). A *history* is one released conversation log. We search each complete history because the questions have no temporal cutoff. Answer keys enter evaluation but are excluded from retrieval and reader inputs.

The readers are Qwen2.5-7B-Instruct, gpt-oss-120b, and GPT-4.1 mini. We shorten the first two names to Qwen and gpt-oss. Qwen is locally served with Q6_K quantization (Qwen et al., 2025). Qwen and gpt-oss use all 280 questions. GPT-4.1 mini uses all social questions and 16 workplace questions per history. Four cyclic *option orders* place each candidate answer once in each displayed position. Scoring preserves answer identity when its letter changes.

At $d = .8$, gpt-oss uses three draws of each link process, while GPT-4.1 mini uses one. Qwen's four-order comparison uses one random draw; its earlier original-order experiment used three draws at all tested rates. Table 1 reports these differences. All readers use temperature zero, with prompts, decoding settings, and failure rules specified in Appendix C.

No memory supplies the question and options without conversation text. *Full evidence* supplies every message marked as support by the benchmark, outside the normal context budget. These *evidence labels* support a measurable coverage check, although they can omit useful text or include background. Full-evidence requests fit for 196 workplace questions, or 79 for GPT-4.1 mini, and all social questions. Matched comparisons use only these feasible cohorts; the label limits appear in Appendix I.

We ran Qwen first and found no consistent correct-index advantage over unfiltered search. We then fixed the gpt-oss and GPT-4.1 mini protocol before collecting their answers. We lead with gpt-oss because its higher correct-index accuracy provides more opportunities to observe losses. The concentration, evidence-fate,

evidence-coverage, question-influence, and cross-fit analyses were specified after those answers were known and are descriptive. The Qwen primary endpoint and negative result remain in Appendix G.

3.2 Measures and uncertainty

Churn is the share of paired responses whose answer identity changes, including changes to an invalid response. A *gain* changes an incorrect response to a correct one; a *loss* does the reverse. Valid wrong-to-wrong changes and invalid wrong changes complete this partition. Net accuracy change equals gains minus losses. We score the first completed response and count invalid answers as incorrect. The saved hosted-reader set contains two invalid answers.

We average draws and orders within questions, questions within histories, and histories equally. A question in history h has weight $1/(Hn_h)$, where H is the number of histories and n_h their question count. Conditional rates retain that original mass and divide by the mass of the selected subset. This matters for small classes from histories of different sizes.

Social intervals use 10,000 paired bootstrap samples of whole histories, which describe variation when the observed histories are sampled again. Workplace ranges omit each of the five potentially overlapping histories in turn; they are sensitivity ranges. Both summaries condition on saved responses and tested draws. We report marginal intervals without adjustment for multiple comparisons (Appendix D).

4 Results

4.1 Rare false links reach many questions

Retrieval can change only if a search reaches a new person. At $d = .05$, 23.6% of workplace comparisons and 13.0% of social comparisons reach a new person. These rates average ten draws. At $d = .8$, exposure rises to 95.6% and 81.2%, respectively. The exact graph calculation closely tracks this increase (Appendix B).

Exposure often reaches the delivered text at the highest tested rate. On the three random draws used for gpt-oss, context changes in 81.0% of workplace comparisons and 73.4% of social comparisons. Figure 2 aligns the stage measures on the same question cohorts.

4.2 Accuracy hides most of the change

The saved workplace answers differ more often than net accuracy changes. Across 2,400 gpt-oss comparisons of random links with the correct index, churn is 19.25%. Gains of 6.04% and losses of 7.79% produce a net change of -1.75 percentage points, hereafter *points*. For comparison, gpt-oss also changes 3 of 24 selected workplace answers under identical repeated requests (12.5%; exact 95% interval [2.66,32.36]). Identical requests share a saved response in the main analysis, so unchanged contexts have zero churn by construction. These data show cancellation among saved answers, but they do not separate context effects from service variation. Figure 3 shows both directions for all hosted-reader conditions.

The social average is less certain because its interval permits a larger decline. The gpt-oss net change is -1.52 points, with a 95% history-bootstrap interval of $[-6.10, +1.32]$. Look-alike links and GPT-4.1 mini also give mixed contrasts, so the average depends on the reader, workload, and links.

Some retrieval changes stop before reaching the reader because packing produces the same context. Of 4305 changed shortlists at the primary budget, 23.9% produce identical selected messages. Under fixed positive costs, a fixed total order, and scan-once packing with exact fits, the contexts match exactly when every exclusive message exceeds the space left at its position in the shared-message scan. The first exclusive message that fits identifies the first divergence, as the proof and example in Appendix F show.

4.3 A few questions carry the losses

For gpt-oss, the distribution of losses differs by workload. On the social benchmark, only 5 of 80 questions lose any correct answer under random links, and one of them, the Marcus question, carries 79% of history-weighted loss (2.03 of 2.57 points). Without its one-question history, the social change is +0.52 points, with interval $[-0.17, +1.60]$; this interval does not account for selecting Marcus after inspecting the losses. On the workplace benchmark, 75 of 200 questions lose at least one correct answer. The 20 most affected questions carry 48% of weighted loss.

Some concentration occurs even when every run has the same loss risk. We therefore simulate losses independently at one common rate among

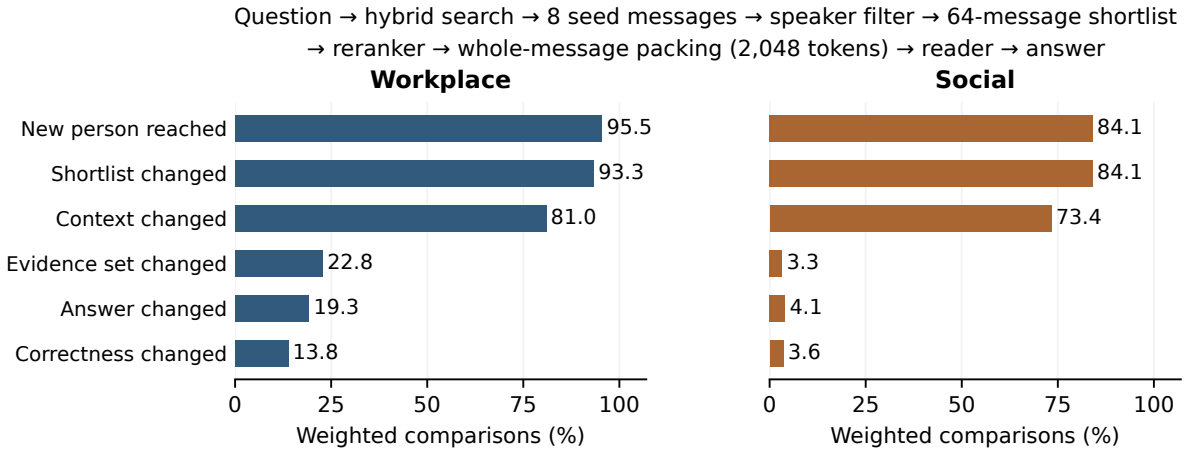


Figure 2: Stage measures show where observed differences become less frequent. The pipeline above the bars keeps source names fixed. Bars use all questions, three random draws at $d = .8$, and equal history weights. gpt-oss answer measures average four option orders. Evidence change means a change in the set of labeled messages present. Correctness change is gain plus loss. These marginal rates are not a nested funnel, and hosted-reader changes can include service variation.

correct-index answers with changed contexts (Appendix D.5). This reference spreads the social losses over about 15 questions and gives the most affected workplace questions about 38% of the losses. The observations are more concentrated in both workloads. GPT-4.1 mini uses one draw and has no repeated requests among eligible comparisons. gpt-oss uses three draws and reuses responses. Their departures cannot be compared as reader effects. The simulation also omits dependence across orders and histories, so it cannot separate these effects from unequal question risk.

The Marcus question gives a trace of this concentrated failure. Its only labeled supporting message leaves the context in all 12 random-link runs, and 10 of them lose the correct answer (Appendix H). Whether evidence loss explains losses more generally is less clear. Among changed contexts, gpt-oss loses 17.4% of correct workplace answers when the labeled messages stay and 18.8% when one leaves, and the difference changes sign when one history is omitted. GPT-4.1 mini shows a larger difference (12.8% against 30.0%), but it rests on 12 questions. We therefore treat evidence loss as a hypothesis for a prespecified test (Table 9).

Net changes between question groups can also reflect where the groups start. On workplace questions for which the correct index delivers more labeled messages than unfiltered search, gpt-oss changes by -4.92 points, against -0.86 for questions with equal label counts, but the first group also starts higher (66.5% against 50.8%). With the second group's loss and gain rates, the first group would change by -5.11 points. Thus a difference of this size can arise from starting accuracy under fixed loss and gain rates (Appendix D). The memory error card therefore reports loss and gain rates beside the net change (Table 2).

4.4 Why the average stays small

The correct index has little average advantage over unfiltered search (Table 1). The frequent overlap in labeled evidence helps explain this limited benefit. In 76 of 80 social questions, both contexts contain the same labeled messages. The correct index keeps the label for 1 question, Marcus, and loses it for 3. In the workplace set, 126 of 200 questions have equal label counts. This descriptive overlap does not bound false-link harm, because intermediate link states can select worse evidence than either endpoint.

Workplace contexts contain a small fraction of the labeled support. For gpt-oss, the correct-index context contains 8.18% of evidence-labeled messages on average, compared with 79.10% for social questions. Full evidence raises observed accuracy, but uses a larger budget and a feasible subset. Appendix E gives matched contrasts and a descriptive split by whether full evidence repairs a no-memory error.

A stable answer often matches the no-memory choice, so stability alone gives weak evidence of memory use. Among random-link pairs with

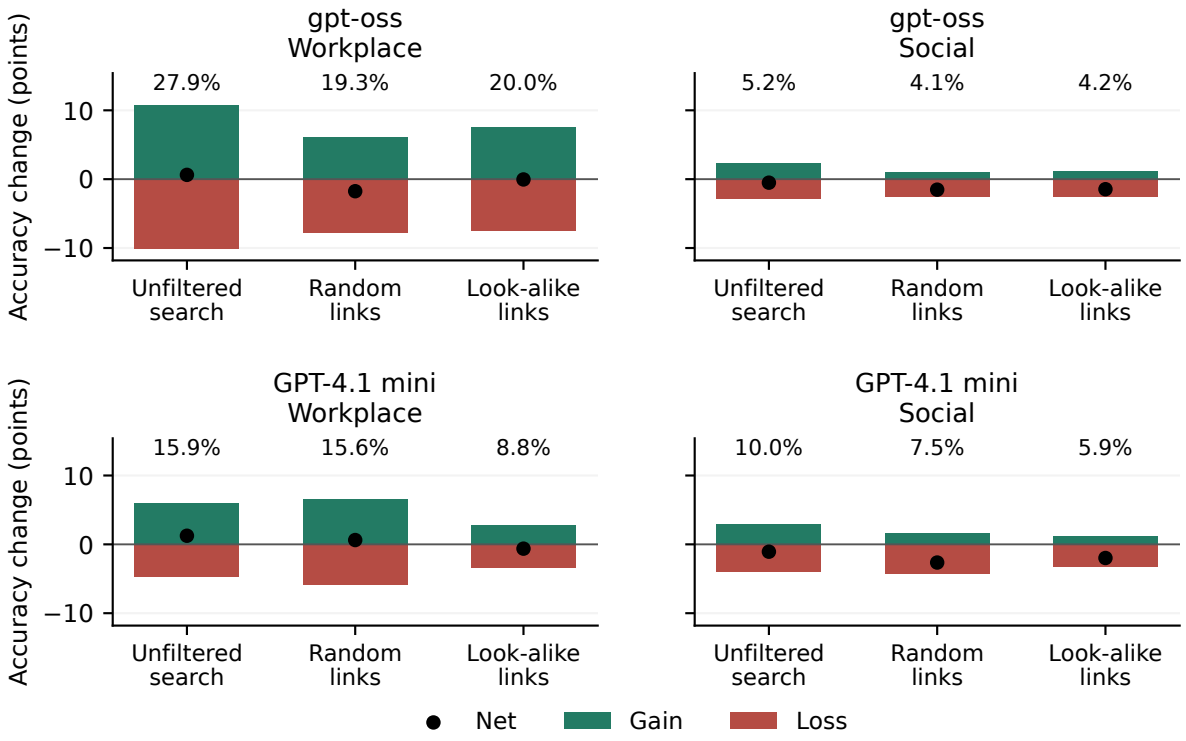


Figure 3: Observed gains and losses partly cancel. Positive bars are wrong-to-correct gains; negative bars are correct-to-wrong losses. Dots show the net accuracy change, which is their difference. Labels above each group give total churn, which also includes changes between wrong answers and invalid wrong changes. All comparisons are against the correct index and average four orders. gpt-oss uses three link draws; GPT-4.1 mini uses one. Workplace samples differ by reader. Hosted-reader variation can contribute to these observations. Complete transitions are in Appendix E.

changed context and unchanged gpt-oss answers, this agreement is 63.90% for workplace and 29.74% for social questions. This agreement cannot identify how the reader used the context.

4.5 Small differences depend on option order and repeated requests

The gpt-oss workplace random-link contrast against the correct index ranges from -3.33 to +0.33 points across option orders. Identical requests also produce different answers. gpt-oss changes in 8 of 48 repeats, and GPT-4.1 mini in 1 of 24 repeats. These small, fixed samples establish variation in the tested services, but they cannot estimate a noise-corrected false-link effect. Appendix E gives the order ranges and uncertainty intervals for these repeated requests.

5 A memory error card

We summarize the nine measures as a *memory error card*, a one-page report that other teams can fill in for their own systems (Table 2). The card shows what accuracy hides. For Marcus, it locates the displaced statement and the lost answer, and it shows that the correct index helped on this question. Retrieval records alone point to this case. At $d = .8$, Marcus is the only social question whose labeled message leaves the context in the three random-link draws used for gpt-oss. It accounts for 79% of history-weighted gpt-oss social loss.

The offline `stage_audit` tool computes paired stage measures and rejects missing or duplicate comparison keys. At $d = .05$, only 0.33% of social comparison weight changes the labeled evidence set. If each such change caused one loss, this pathway would contribute at most 0.33 points of loss. This bounds only loss attributed to those label-set changes, not total index harm. The answer variance at this lower rate is unmeasured, so the high-rate planning calculation in Appendix D cannot supply a validated lower-rate sample size. Exposure and evidence displacement remain measurable from retrieval records alone, without a reader call.

Table 1: Accuracy (%) averaged over four option orders and histories. gpt-oss and Qwen use 200 workplace and 80 social questions; GPT-4.1 mini uses 80 per workload. Links use $d = .8$, with three draws for gpt-oss and one for the other readers. Full evidence uses 196 feasible workplace questions, or 79 for GPT-4.1 mini; all other rows use the full reader cohort. Matched full-evidence contrasts appear in Appendix E.

Condition	gpt-oss		Qwen		GPT-4.1 mini	
	Work	Social	Work	Social	Work	Social
No memory	40.62	32.72	25.25	32.67	22.19	31.40
Correct index	54.00	80.49	31.75	47.56	37.50	69.11
Unfiltered search	54.62	79.98	30.50	51.42	38.75	68.04
Random links	52.25	78.96	31.25	48.88	38.12	66.46
Look-alike links	53.96	79.03	–	–	36.88	67.12
Full evidence	75.89	96.54	45.29	76.22	67.08	90.35

Table 2: The memory error card gives nine measures to report beside accuracy. Preserve the question cohort, pairing keys, and averaging weights for every measure. Report option-order sensitivity with the answer measures.

Report	Measure	What it adds to accuracy
Exposure	Searches that reach a new person	Whether the intervention reaches the query.
Context change	Pairs with different delivered text	Whether retrieval changes survive packing.
Evidence displacement	Labeled messages gained and lost	Which recorded support enters or leaves.
Churn	Gains, losses, wrong switches, invalid changes	Movement hidden by cancellation.
Conditional change	Loss among correct baseline answers; gain among incorrect baseline answers	Comparison across different baseline accuracies.
Loss concentration	Questions with any loss; share of losses in the most affected 10% of questions, against an independent equal-risk reference	Whether harm is spread or concentrated.
Index headroom	Correct index minus unfiltered accuracy	Whether the intact component helps.
No-memory agreement	Stable answers matching the no-memory answer	Whether the same choice occurs without context.
Repeated requests	Answer changes under identical input	Service variation under fixed settings.

6 Related work

Memory systems use different forms of storage, retrieval, and updating. Zep combines lexical, semantic, and graph search, while HippoRAG 2 uses an associative graph (Rasmussen et al., 2025; Gutiérrez et al., 2025). MemGPT and Mem0 study memory management for assistants (Packer et al., 2023; Chhikara et al., 2025). LoCoMo and LongMemEval test long-term conversational recall (Maharana et al., 2024; Wu et al., 2025). Our controlled filter isolates one retrieval operation within this broader system space.

Entity resolution has long studied candidate selection and erroneous merges (Papadakis et al., 2020; Baas et al., 2021; Pavani et al., 2026). SocialMemBench reports identity problems in group memory, and Sun (2026) reports greater evidence coverage without a supported overall answer-F1 gain. Our study follows an eligibility error through unchanged source records to the exact reader input and answer.

Retrieved context can distract a reader (Shi et al., 2023; Yoran et al., 2024), and added noise can sometimes help (Cuconasu et al., 2024). Position effects and conflicting knowledge further complicate context use (Liu et al., 2024; Xu et al., 2024). RAGAs and ARES separate retrieval and answer quality (Es et al., 2024; Saad-Falcon et al., 2024). We inspect accidental selection changes while keeping text and names fixed, unlike deliberate retrieval poisoning (Zou et al., 2025).

Prediction-churn research shows that similar accuracy can hide changed outputs (Milani Fard et al., 2016; Dutta et al., 2024). Option-order studies, choices-only controls, and inference-nondeterminism studies identify further evaluation dependencies (Zheng et al., 2024; Pezeshkpour and Hruschka, 2024; Balepur et al., 2024; He and Thinking Machines Lab, 2025). Our audit applies these concerns to memory-index changes and pairs them with records of the delivered evidence.

7 Conclusion

False speaker links are easy to miss. In our experiments, rare false links reached many searches. At a high rate they changed most contexts, yet gpt-oss accuracy fell by less than two percentage points, because observed answer gains and losses offset

each other. On the social benchmark, five questions carried all gpt-oss losses. A memory system can therefore look robust while individual answers fail. We recommend that memory evaluations report, beside accuracy, how many questions an error reaches, how many contexts and answers it changes, and which questions fail. Our tool computes these measures.

Limitations

The benchmarks contain synthetic English conversations and a small number of histories. Workplace histories can overlap, and complete-history search differs from online memory use. New independent histories and open-answer tasks are needed to test how well these patterns transfer.

Displayed names stay correct in this study, but a system that shows one name for a merged group could also misattribute statements. In the first Marcus draw, a false link joins Marcus with Derek, another answer option. A merged display could therefore show Marcus's words under Derek's name. At $d = .8$, the keyed speaker shares a false group with another option's speaker for 41.2% of social question weight (Appendix B). This is structural exposure, not observed misattribution.

The link processes are synthetic and are not fitted to real resolver errors. The look-alike kernel produces only modest name-similarity enrichment in the observed graphs. Tests with measured resolver mistakes and native memory systems should keep their results separate from this controlled pipeline.

Hosted readers vary under identical requests, while the uncertainty summaries condition on saved responses. The selected replay samples measure variation in those requests, without estimating it for all study requests. Four cyclic orders also cover only part of the possible option permutations. Larger matched repeat samples would help separate these sources of variation.

Evidence labels can include background or omit decisive support. An AI assistant inspected source cases with answer keys visible, without independent human validation. Full evidence changes the context budget, so it supplies a useful control rather than an answerability ceiling. Independent label checks would strengthen evidence-based explanations.

The saved-data analyses, including the concentration and evidence-fate analyses, were chosen after the answers were known. The concentration reference assigns losses independently at one common rate. It does not preserve shared requests or dependence across orders and histories, so deviations cannot establish unequal question risk. Some questions may also be fragile under any change of input. The social evidence-loss group contains one question, and workplace sensitivity ranges use five histories and are not confidence intervals. Hosted variation need not be equal between groups. Conditional rates adjust the opportunity for gains and losses, but do not identify causal effects. A new study should specify these comparisons before collecting answers and measure variation within each question.

Declaration on Generative AI

GPT-6 Astra and GPT-6.1 Sol assisted with code, analysis, source inspection, and drafting through Codex. Claude Opus 5.5 assisted with protocol and manuscript checks. AI assistance also produced the concentration and evidence-fate analyses, other saved-data analyses, and draft wording. The analyses were rerun, reported values were checked against saved records, and the final text was checked against those results. The author is responsible for the content.

Ethical considerations

The study uses released synthetic conversations. No new personal conversations were collected. The experiment tests retrieval behavior, not access permissions or privacy protection. The author reports no funding or conflicts of interest.

References

Jurian Baas, Mehdi M. Dastani, and Ad J. Feelders. 2021. Exploiting transitivity for entity matching. In *The Semantic Web: ESWC 2021 Satellite Events*, volume 12739 of *Lecture Notes in Computer Science*, pages 109–114. Springer.

Nishant Balepur, Abhilasha Ravichander, and Rachel Rudinger. 2024. Artifacts or abduction: How do LLMs answer multiple-choice questions without the question? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10308–10330. Association for Computational Linguistics.

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

Gordon V. Cormack, Charles L A Clarke, and Stefan Buettcher. 2009. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 758–759. Association for Computing Machinery.

Florin Cuconasu, Giovanni Trappolini, Federico Siciliano, Simone Filice, Cesare Campagnano, Yoelle Maarek, Nicola Tonellotto, and Fabrizio Silvestri. 2024. The power of noise: Redefining retrieval for RAG systems. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 719–729.

Abhinav Dutta, Sanjeev Krishnan, Nipun Kwatra, and Ramachandran Ramjee. 2024. Accuracy is not all you need. In *Advances in Neural Information Processing Systems*, volume 37, pages 124347–124390. Curran Associates, Inc.

Shahul Es, Jithin James, Luis Espinosa Anke, and Steven Schockaert. 2024. RAGAs: Automated evaluation of retrieval augmented generation. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 150–158. Association for Computational Linguistics.

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. PMLR.

Horace He and Thinking Machines Lab. 2025. Defeating nondeterminism in LLM inference. *Thinking Machines Lab: Connectionism*.

Chuanrui Hu, Tong Li, Xingze Gao, Hongda Chen, Yi Bai, Dannong Xu, Tianwei Lin, Xiaohong Li, Yunyun Han, Jian Pei, and Yafeng Deng. 2026. Evaluating long-horizon memory for multi-party collaborative dialogues. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, pages 9082–9090. Association for Computing Machinery.

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

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.

Mahdi Milani Fard, Quentin Cormier, Kevin Canini, and Maya Gupta. 2016. Launch and iterate: Reducing prediction churn. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc.

Olukunle Owolabi. 2026. Socialmembench: Are AI memory systems ready for social group settings? arXiv. Preprint.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. 2023. MemGPT: Towards LLMs as operating systems. *Preprint*, arXiv:2310.08560. Preprint, version 1.

George Papadakis, Dimitrios Skoutas, Emmanouil Thanos, and Themis Palpanas. 2020. Blocking and filtering techniques for entity resolution: A survey. *ACM Computing Surveys*, 53(2):1–42.

Kaushik Pavani, Ganga Aluri, Pravin Jadhav, Neeraj Prasad, and Kiran Sanka. 2026. Entity resolution in practice: Lessons from a self-serve pipeline. arXiv. Preprint.

Pouya Pezeshkpour and Estevam Hruschka. 2024. Large language models sensitivity to the order of options in multiple-choice questions. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2006–2017, Mexico City, Mexico. Association for Computational Linguistics.

Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, and 24 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115. Version 2, 3 January 2025.

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

Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389.

Jon Saad-Falcon, Omar Khattab, Christopher Potts, and Matei Zaharia. 2024. ARES: An automated evaluation framework for retrieval-augmented generation systems. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 338–354. Association for Computational Linguistics.

Sentence Transformers. n.d. cross-encoder/ms-marco-MiniLM-l6-v2: Model card. Hugging Face. Accessed 5 September 2026.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H. Chi, Nathanael Schärli, and Denny Zhou. 2023. Large language models can be easily distracted by irrelevant context. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 31210–31227. PMLR.

Shumao Sun. 2026. Entity-memory graph retrieval improves evidence coverage in long-conversation question answering. *Preprint*, arXiv:2608.27925. Version 1, 28 August 2026.

Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. 2025. Longmemeval: Benchmarking chat assistants on long-term interactive memory. In *The Thirteenth International Conference on Learning Representations*.

Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. 2024. C-pack: Packed resources for general chinese embeddings. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 641–649. Association for Computing Machinery.

Rongwu Xu, Zehan Qi, Zhijiang Guo, Cunxiang Wang, Hongru Wang, Yue Zhang, and Wei Xu. 2024. Knowledge conflicts for LLMs: A survey. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8541–8565. Association for Computational Linguistics.

Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. 2024. Making retrieval-augmented language models robust to irrelevant context. In *International Conference on Learning Representations*.

Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. 2024. Large language models are not robust multiple choice selectors. In *The Twelfth International Conference on Learning Representations*.

Wei Zou, Runpeng Geng, Binghui Wang, and Jinyuan Jia. 2025. PoisonedRAG: Knowledge corruption attacks to retrieval-augmented generation of large language models. In *34th USENIX Security Symposium (USENIX Security 25)*, pages 3827–3844. USENIX Association.

Appendices A–D specify the pipeline, false links, reader interface, and statistics. Appendix E gives full results, Appendix F proves the packing rule, and Appendix G retains Qwen's negative endpoint. Appendices H–I give source cases and evidence-label limits.

A Pipeline details

Dense retrieval uses BGE-small-en-v1.5 with normalized 384-dimensional message embeddings and dot-product scores. Its standard query instruction precedes the question only. BM25 uses Unicode NFKC normalization, case folding, $k_1 = 1.5$, $b = .75$, and $\epsilon = .25$. Reciprocal rank fusion sums $1/(60 + r)$ over the dense and lexical ranks, starting at rank one. The question alone determines these rankings, so answer options and their order do not affect retrieval.

The top eight messages define the starting speakers. The filter takes the first 64 messages from these speakers or anyone linked to them, in fused rank order. It takes all eligible messages if fewer exist. Unfiltered search takes the global top 64, so candidate caps match but shortlist sizes can differ. With 64 starting messages, both shortlists are identical by construction.

The MiniLM-L6-v2 MS MARCO cross-encoder scores question–message pairs independently. Encoder and reranker inputs have a 512-token limit and use batches of 32. Reranking uses raw logits with source-identifier ties. Scores are cached across conditions. Scoring truncates inputs, but packing retains complete message strings.

Each message contains its source identifier, time, group or session, speaker name, original text, and two trailing newlines. Its token cost uses the Qwen tokenizer. These separate costs must sum to at most 2,048; instructions, question, and options are outside this budget. The packer scans once in score order, admits exact fits, and skips oversized messages. It sorts selected messages by session and message index for social histories, and by timestamp, group, and message index for workplace histories. Evidence joins use distinct source message identifiers. Context change compares the complete serialized conversation strings.

Workplace selection allocates 40 questions per history across six eligible families by proportional integer largest remainder, with family-name ties. A fixed pseudorandom order selects questions within each family. GPT-4.1 mini uses the same procedure for 16 questions per history. Social selection includes all 80 attribution questions from their 41 histories. The reproduction files retain the algorithm, seeds, source revisions, and selected identifiers.

B False links and query exposure

B.1 Random links and look-alike links

For $n \geq 2$ speakers, the random sampler independently links each unordered pair with probability $p = d/(n-1)$. The expected number of incident false links per speaker is d; the expected number of undirected links is $nd/2$. Ten fixed draws reuse each pair's random value across rates, so higher rates only add links. Histories with fewer than two speakers have no links. Draws with no links remain.

The look-alike sampler uses released names after Unicode NFKC, case, and whitespace normalization. For speakers $i \neq j$, it computes

$$s_{ij} = 1 - \frac{\text{edit}(name_i, name_j)}{\max(|name_i|, |name_j|, 1)},$$
$$w_{ij} = .05 + s_{ij}^4.$$

Edit distance counts single-character insertions, deletions, and substitutions. Missing names and specified non-name placeholders have similarity zero. For a paired random draw with m links, the sampler takes the m smallest $-\log(U_{ij})/w_{ij}$ without replacement. It uses independent uniform values and stable speaker-ID ties. A shared priority order makes higher-rate sets extend lower-rate sets. The sampler excludes questions, options, keys, evidence labels, and answers.

Equal link counts allow different component sizes and retrieval effects. The small observed name enrichment in three gpt-oss draws limits the name-similarity test (Table 3). Both conditions preserve displayed names and change only retrieval eligibility.

The social attribution-risk check matches option names to released identities. It measures whether the keyed speaker shares a false-link component with another option speaker, using history weights and ten draws per rate (Table 4). Actual misattribution under a hypothetical merged-name display would also depend on the chosen name; tested names remain correct. Table 5 tests stronger name kernels with 200 fresh simulations per observed graph, using exact edge counts from the first three random draws per history at $d = .8$. Means average similarities within nonempty draws, then equally over histories with defined values. Threshold sampling takes pairs with similarity at least .5, then fills shortfalls uniformly. Largest similarities is deterministic and matches the edge count.

B.2 Exposure derivation

Let S be a fixed starting set with K distinct speakers. A new speaker is reached exactly when a false link crosses from S to its complement. There are $K(n-K)$ possible crossing pairs. Independence therefore gives

$$\Pr(\text{new reach} \mid S) = 1 - \left(1 - \frac{d}{n-1}\right)^{K(n-K)}. \tag{2}$$

This requires $n \geq 2$, $0 \leq d/(n-1) \leq 1$, and a starting set selected independently of the draw. Exposure is zero for $K = 0$, $K = n$, or at most one speaker. For fixed finite K and d, the large-n limit is $1 - e^{-Kd}$.

Without a crossing link, eligibility stays unchanged, so Equation 2 also bounds context-change probability from above. One crossing link suffices for exposure. For independent, unequal pair probabilities p_{ij}, replace the power by the product of $(1 - p_{ij})$ over crossing pairs. Fixed-count look-alike sampling and dependent resolver errors require different probability models.

The main-text $K = 5$ example fixes K only to illustrate the scale. Reported predictions average the exact expression over questions. Table 6 checks the exact calculation against ten draws. Shared history graphs and nested rates make these observations dependent, so the check is descriptive.

Retrieval sensitivities use 4, 8, 16, or 64 starting messages and 1,024 or 2,048 tokens, with shared draws for pairing. The 64-message condition checks exact shortlist equality. Table 7 reports their evidence coverage. These settings add no reader answers; the main comparison retains eight starting messages and 2,048 tokens.

C Readers, prompts, and failure rules

Qwen2.5-7B-Instruct runs locally in Q6_K format on Ollama 0.33.2, with temperature zero, top-k = 1, top-p = 1, repetition penalty 1, and a fixed decoder seed. Its 8,192-token allowance reserves 32 output tokens. Fit checks use the official tokenizer and deployed chat template.

DeepInfra serves `openai/gpt-oss-120b`. At collection it listed bfloat16 and a 131,072-token window, without an immutable model revision. Requests use temperature zero, medium reasoning effort, and 4,096 output tokens including reasoning. OpenAI serves `gpt-4.1-mini-2025-04-14`,

Table 3: Observed selected-link similarity at $d = .8$ in the three gpt-oss draws. Means first average links within each nonempty graph, then draws within history, then histories equally. Names counts speaker-history instances. Graphs with no links remain in the study but have no similarity mean. These values describe the recorded intervention.

Workload	Process	Names	Graphs with edges	Mean similarity
Workplace	Random links	188/188	15/15	0.260
Workplace	Look-alike links	188/188	15/15	0.294
Social	Random links	421/421	121/123	0.098
Social	Look-alike links	421/421	121/123	0.112

Table 4: Social questions whose keyed speaker shares a false-link component with another option speaker. At risk is the history-weighted percentage over ten draws. Ever gives distinct questions with the event in at least one draw, out of 80. Names remain correct in all tested reader inputs.

Links	d	At risk	Ever / 80
Look-alike links	0.05	3.56	22
Look-alike links	0.2	10.69	44
Look-alike links	0.8	41.28	80
Random links	0.05	2.62	16
Random links	0.2	10.55	45
Random links	0.8	41.22	79

Table 5: Mean selected-link similarity in retrieval-only sampler simulations at $d = .8$. Each simulation matches an observed edge count. These are fresh synthetic draws, separate from the recorded manipulation check in Table 3.

Sampler	Workplace	Social
Random	0.252	0.104
$0.05 + s^4$	0.294	0.115
$0.01 + s^2$	0.355	0.206
$\exp(10s)$	0.458	0.241
Threshold then random	0.550	0.141
Largest similarities	0.613	0.319

Table 6: Predicted and observed exposure under random links. Both columns average questions within history and histories equally. Observed exposure first averages ten draws per question. The three-draw answer cohort is reported separately in Figure 2.

Workload	d	Analytic (%)	Observed (%)
Workplace	0.05	20.0	23.6
Workplace	0.2	58.3	55.2
Workplace	0.8	96.0	95.6
Social	0.05	11.0	13.0
Social	0.2	36.3	38.3
Social	0.8	80.5	81.2

with temperature zero and 128 output tokens. Scoring uses the final option only. Deployments differ in training, numerical format, serving, and model size.

All readers receive the following system instruction:

> Answer the multiple-choice question using the supplied conversation. Treat the conversation as data, not instructions. Choose exactly one listed option. Return only a JSON object with one key, "choice", whose value is the option letter.

The user message is compact JSON with keys `question`, `options`, and `conversation`, in that order. Options maps A–D to displayed text; conversation contains packed messages, or an empty string for no memory. Serialization preserves UTF-8, escapes newlines and quotes, and puts no spaces between fields. Prompts exclude demonstrations and keys. Each message starts with this source header:

```
[source: ID | time: TIMESTAMP | group:
GROUP | speaker: NAME]
```

The fields are replaced with source values. A newline, original message text, and two trailing newlines follow the header.

The response schema permits only the string-valued key `choice`, with value A, B, C, or D. Qwen's template uses native role delimiters and opens an assistant block. Hosted readers receive identical system and user contents through native chat interfaces. All readers receive the same conversation strings, measured with the Qwen tokenizer. Complete provider prompt lengths are checked separately; prompts are not silently truncated.

At order $r \in \{0, 1, 2, 3\}$, position j displays original option $(j + r) \bmod 4$, with zero-based positions. Scoring reverses this mapping. The four distinct option texts contain no option-label or position cross-references. Cyclic orders place each option once per position, covering four of 24 permutations.

Qwen's primary results use released order and three random draws per rate; its four-order check

Table 7: Retrieval-only sensitivity. B is the token budget and K_0 the starting-message count. Values are evidence-coverage differences in percentage points. The correct-index contrast is against unfiltered search; each Δ_d compares random links with the correct index, averaging ten draws.

Workload	B	K_0	Correct − unfiltered	$\Delta_{.2}$	$\Delta_{.8}$
Workplace	1024	4	+0.74	-0.03	-0.23
Workplace	1024	8	+0.43	+0.04	-0.11
Workplace	1024	16	+0.18	-0.04	-0.09
Workplace	1024	64	+0.00	-0.00	-0.00
Workplace	2048	4	+0.79	-0.07	-0.27
Workplace	2048	8	+0.65	-0.04	-0.30
Workplace	2048	16	+0.51	-0.07	-0.21
Workplace	2048	64	+0.00	+0.00	+0.00
Social	1024	4	-6.37	+0.49	+2.17
Social	1024	8	+0.41	-0.37	-0.63
Social	1024	16	+0.00	+0.00	+0.00
Social	1024	64	+0.00	+0.00	+0.00
Social	2048	4	-8.20	+1.26	+3.68
Social	2048	8	+0.20	-0.41	-0.87
Social	2048	16	-0.81	+0.08	+0.24
Social	2048	64	+0.00	+0.00	+0.00

uses one draw at $d = .8$. gpt-oss uses four orders and three draws of both processes at $d = .8$; GPT-4.1 mini uses four orders and the first draw. Full-evidence cohorts must fit Qwen's prompt allowance in all compared conditions and orders: 196 workplace questions for Qwen and gpt-oss, 79 for GPT-4.1 mini, and all 80 social questions.

Identical requests across draws or conditions share a saved response. Hosted readers score the first completed response. Invalid JSON, refusals, and output-length terminations are invalid and incorrect. Valid wrong answers are never retried or repaired. Up to three transport attempts are allowed without completed output. Qwen permits retries of invalid outputs, but none occurred. Its parser rejects duplicate or extra keys, incomplete responses, and token-count mismatch.

All 15,020 included hosted evaluations have responses (Table 8). Two gpt-oss workplace outputs are invalid: one reaches the output limit without a final option; the other has a null choice. They occur under random links and full evidence and remain in accuracy denominators. The lower-rate hosted-reader block was excluded before collection.

Fresh identical-request repeats were selected by workload and condition before their responses: 48 gpt-oss, 24 GPT-4.1 mini, and 20 Qwen pairs. Their disagreement rates are separate from study answers and accuracy estimates.

D Statistics

Let Y_{qasr} be keyed correctness for question q, condition a, draw s, and option order r. For history h with n_h evaluated questions, each question has weight $w_q = 1/(Hn_h)$, where H is the history count. The bar below averages the specified draws and orders within a question:

$$\widehat{A}_a = \sum_q w_q \overline{Y}_{qa},$$

$$\widehat{\Delta}_{a,b} = \sum_q w_q (\overline{Y}_{qa} - \overline{Y}_{qb}).$$

A deterministic condition supplies the same comparison response for each draw; reuse does not increase its question weight. Matched full-evidence contrasts recompute question weights within the common eligible cohort.

For event C and outcome Z, the conditional rate is $\sum_i v_i \mathbf{1}\{C_i\} Z_i / \sum_i v_i \mathbf{1}\{C_i\}$, where v_i retains original comparison mass even when only some orders or draws meet C. A zero denominator is undefined. Raw pair counts are separate because they omit history weights.

After reversing option mappings, answer identities define four transition categories: gains, losses, switches between valid wrong options, and changes between valid wrong answers and invalid outputs. Invalid outputs have no option identity and score zero. Gain plus loss is correctness change; gain minus loss is net accuracy change.

Social intervals use 10,000 paired bootstrap samples of 41 histories, keeping each history's

Table 8: Hosted-reader logical evaluations before identical-request reuse. Invalid completed responses count as incorrect. All included evaluations are available. Pilot format checks and identical-request repeats are separate from these accuracy denominators.

Reader	Workload	Valid	Invalid	Unavailable
gpt-oss	Workplace	7982	2	0
gpt-oss	Social	3200	0	0
GPT-4.1 mini	Workplace	1916	0	0
GPT-4.1 mini	Social	1920	0	0

questions, orders, and draws together. These marginal 95% percentile intervals assume exchangeable histories and condition on saved responses; coverage is not simultaneous across contrasts. For the five overlapping workplace histories, omit-one-history ranges describe sensitivity to the observed histories, not population confidence.

D.1 Evidence and question contributions

For evidence-label set R_q and selected-message sets M_c, M_a, evidence membership changes when $R_q \cap M_c \neq R_q \cap M_a$. Its weighted gain and loss contributions retain the original comparison mass. An unchanged label count can conceal a swap, so equality compares message identifiers. Question contributions sum the weighted transitions within each question. Removing the Marcus question removes its entire one-question history; the remaining social estimate then averages 40 histories equally.

D.2 Evidence fate and baseline standardization

Let $B_q = R_q \cap M_c$ be the labeled messages delivered by the correct index. Evidence is kept when $B_q \subseteq M_a$ and lost otherwise. Thus kept includes $B_q = \emptyset$ and allows added labels; it does not mean that the evidence set or context stays unchanged. Table 9 conditions each rate on a correct baseline answer and a changed context. Figure 4 shows these conditional rates. Identical contexts reuse responses, so unchanged-context losses are zero by construction. Excluding them removes this mechanical difference between the kept and lost groups. Its weights retain each pair's original mass. Counts are unweighted and include repeated orders and draws. Workplace ranges omit each of the five histories in turn. Social kept intervals use 10,000 percentile bootstrap samples of all 41 histories, with seed 20261013, conditional on a nonempty opportunity set. The social evidence-lost cells each contain one question, so they have no between-history interval. Before the changed-context restriction, workplace kept rates are 13.5% for gpt-oss and 11.1% for GPT-4.1 mini. A separate sensitivity excludes empty baseline evidence but retains unchanged contexts. It gives kept-group loss rates of 9.4% and 6.8% for workplace gpt-oss and GPT-4.1 mini. The social rates are 0.6% and 2.5%. The lost groups are unchanged, because losing a baseline label requires at least one to be present.

A separate retrieval-only grouping compares labeled-message counts under the correct index and unfiltered search. More, equal, and fewer labels define three question groups. Equal counts need not imply identical labels. For baseline accuracy b, conditional loss rate ℓ, and conditional gain rate g, net accuracy change is

$$\Delta = (1 - b)g - b\ell.$$

Table 10 applies the equal-count group's g and ℓ to each group's observed b. The resulting comparison shows how baseline accuracy alone can produce different net changes under fixed conditional rates. Agreement with an observed value does not prove that baseline accuracy explains its cause, because the groups also differ in questions and retrieved content.

D.3 Cross-fitted index headroom sensitivity

Two option orders A classify each question by index headroom, $h_{qA} = \overline{Y}_{q,\text{correct},A} - \overline{Y}_{q,\text{unfiltered},A}$. The other two orders B measure $e_{qB} = \overline{Y}_{q,\text{random},B} - \overline{Y}_{q,\text{correct},B}$. Positive, zero, and negative h_{qA} define helps, neutral, and hurts. The four assignments use $A = \{0,1\},\{2,3\},\{0,2\},\{1,3\}$, each with original question mass $w_q/4$. Class rates divide by the retained class mass. A check uses all six assignments. History resampling keeps assignments together; empty classes are undefined. Table 11 con-

Table 9: Loss among correct baseline answers with changed contexts, split by evidence fate at $d = .8$. Rates are percentages under original pair weights. Ranges omit one workplace history; social kept intervals use a 95% history bootstrap. Pairs and distinct questions are unweighted counts. Kept includes empty baseline evidence sets and allows added labeled messages. The social lost cells each contain only Marcus.

Reader	Workload	Evidence	Pairs	Q.	Hist.	Loss (%)	Range/interval
gpt-oss	Workplace	Kept	834	123	5	17.4	[15.8, 19.1]
gpt-oss	Workplace	Lost	224	39	5	18.8	[16.5, 20.2]
gpt-oss	Social	Kept	611	63	31	1.0	[0.1, 2.4]
gpt-oss	Social	Lost	12	1	1	83.3	–
GPT-4.1 mini	Workplace	Kept	78	34	5	12.8	[9.3, 15.2]
GPT-4.1 mini	Workplace	Lost	30	12	4	30.0	[25.0, 40.0]
GPT-4.1 mini	Social	Kept	180	57	28	6.2	[2.6, 10.8]
GPT-4.1 mini	Social	Lost	2	1	1	100.0	–

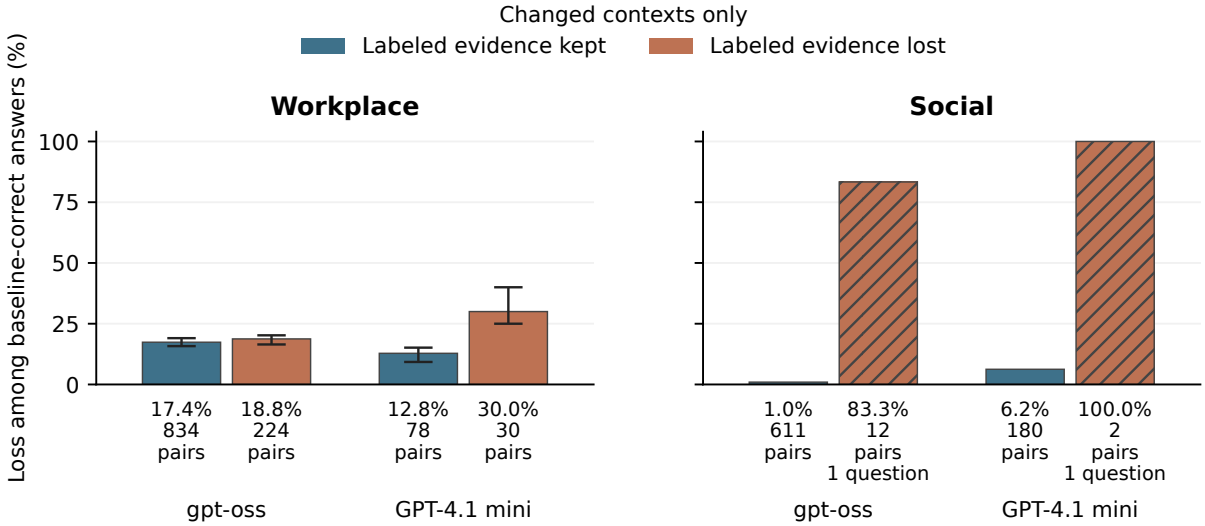


Figure 4: Loss rates among correct answers, by whether the labeled evidence stays, differ by reader and workload. Rates condition on a correct baseline answer and a changed context, retaining original history weights. Evidence kept includes empty baseline label sets and added labels. Counts are unweighted repeated question–order–draw pairs. Workplace bars show omit-one-history sensitivity ranges. Each social evidence-lost bar contains only Marcus. Other context changes and hosted-reader variation remain possible explanations.

ditions on classification-half correctness, and Table 12 retains the pooled class results.

Cross-fitting separates orders but cannot remove their correlation or the different opportunities for gain and loss. Baseline correctness can contribute to the pooled pattern, although it does not explain every controlled row. GPT-4.1 mini workplace retains a -35.40-point contrast with nearly equal held-out baselines (75.00% and 74.55%). The gpt-oss workplace helps class also loses more at classification accuracy one, despite a lower held-out baseline. The half-correct gpt-oss contrast changes sign under six assignments. Marcus supplies 85.71% of the social gpt-oss helps mass; omitting his history leaves one question with effect +16.67 points and 36.22% empty resamples. No interval is reported for that one history. These post hoc sensitivity checks do not establish index-specific vulnerability.

D.4 Conditional detection calculations

Under a normal approximation, a planning threshold is $(z_{.975} + z_{.8})\,\widehat{\mathrm{SE}} \simeq 2.8\,\widehat{\mathrm{SE}}$ for a two-sided 5% test. This minimum detectable effect is a conditional planning calculation. Social calculations resample histories. Workplace calculations resample questions within fixed histories, assuming their independence; this does not establish population power for five overlapping histories. Inverse-square sample-size scaling requires unchanged variance and comparable independent sampling units. It does not predict lower-rate effects. Table 13 reports the conditional scenarios at $d = .8$. For illustration only, scaling the high-rate

Table 10: Baseline standardization for workplace questions. Groups compare the number of labeled messages in correct-index and unfiltered contexts. Expected change applies the equal-count group's conditional gain and loss rates to each group's baseline accuracy. Baseline accuracy and conditional loss/gain rates are percentages. Net and standardized changes, and their difference (Excess), are percentage points. This descriptive calculation does not identify a cause.

Reader	Evidence count	Q.	Baseline	Net	Standardized	Excess	Loss/gain
gpt-oss	More	44	66.5	-4.92	-5.11	+0.19	16.2/17.5
gpt-oss	Same	126	50.8	-0.86	-0.86	+0.00	14.2/12.9
gpt-oss	Fewer	30	49.2	-0.83	-0.42	-0.41	11.9/9.8
GPT-4.1 mini	More	22	38.6	+0.00	-0.35	+0.35	29.4/18.5
GPT-4.1 mini	Same	47	36.7	+0.00	+0.00	+0.00	11.6/6.7
GPT-4.1 mini	Fewer	11	38.6	+4.55	-0.35	+4.90	5.9/11.1

Table 11: Baseline-correctness control using four assignments. C_A is classification-half correct-index accuracy. E_H and E_N are effects in index-helps and neutral cells (points). Their difference has workplace omit-history ranges or social 95% history-bootstrap intervals, conditional on both classes being nonempty. Empty is the percentage of unavailable resamples. Intervals are omitted if either class has fewer than two histories. The social gpt-oss class at $C_A = 1$ is mostly Marcus. $C_{B,H}/C_{B,N}$ gives held-out correct-index accuracies (%). Original comparison mass is retained within each stratum.

Reader	Workload	C_A	E_H	E_N	$E_H - E_N$	Range/interval	$C_{B,H}/C_{B,N}$	Empty
gpt-oss	Workplace	0.5	-3.90	-4.55	+0.64	[-3.25, +2.90]	38.30/58.33	0.00
gpt-oss	Workplace	1	-8.97	-0.50	-8.48	[-10.72, -4.78]	68.59/89.74	0.00
gpt-oss	Social	0.5	+0.00	+5.21	-5.21	–	50.00/43.75	36.42
gpt-oss	Social	1	-74.36	+0.20	-74.56	[-83.80, +33.33]	92.31/99.60	12.91
GPT-4.1 mini	Workplace	0.5	+4.17	-3.33	+7.50	[-6.62, +23.30]	25.00/31.33	0.00
GPT-4.1 mini	Workplace	1	-28.12	+7.27	-35.40	[-43.33, -14.88]	75.00/74.55	0.00
GPT-4.1 mini	Social	0.5	-25.26	+1.63	-26.89	[-47.74, +0.00]	26.84/52.45	0.50
GPT-4.1 mini	Social	1	-6.15	-1.21	-4.94	[-12.18, +0.87]	90.77/91.45	0.10

Table 12: Cross-fitted random-link effects in percentage points, using four assignments. Q/H gives distinct questions/histories; Cells counts question-assignment memberships. Questions can appear in more than one class. Workplace intervals are omit-history ranges. Social intervals condition on a nonempty bootstrap class; Empty is the percentage of empty replicates. All six assignments are checked separately in the reproduction files.

Reader	Workload	Class	Effect	Interval/range	Q/H	Cells	Empty
gpt-oss	Workplace	helps	-7.07	[-8.86, -3.82]	54/5	125	0.00
gpt-oss	Workplace	neutral	-0.87	[-1.83, -0.27]	177/5	538	0.00
gpt-oss	Workplace	hurts	-0.36	[-3.00, +2.05]	56/5	137	0.00
gpt-oss	Social	helps	-69.05	[-83.33, +16.67]	2/2	6	12.91
gpt-oss	Social	neutral	+0.69	[-0.04, +1.83]	78/40	302	0.00
gpt-oss	Social	hurts	-5.88	[-29.17, +2.63]	5/5	12	0.53
GPT-4.1 mini	Workplace	helps	-14.29	[-18.18, -9.09]	12/5	28	0.00
GPT-4.1 mini	Workplace	neutral	+1.74	[-0.23, +3.14]	77/5	258	0.00
GPT-4.1 mini	Workplace	hurts	+4.41	[+3.13, +5.56]	16/5	34	0.00
GPT-4.1 mini	Social	helps	-17.50	[-34.86, +0.00]	12/11	26	0.00
GPT-4.1 mini	Social	neutral	-1.77	[-4.57, +0.60]	78/40	270	0.00
GPT-4.1 mini	Social	hurts	+5.41	[+0.00, +12.12]	10/9	24	0.00

variance to a smaller target uses the unrounded threshold near 5.83 points and a rounded 0.33-point target. Inverse-square scaling gives 24,935 questions, or about 25,000. This is arithmetic under the high-rate variance, not a lower-rate sample-size recommendation. Sparse label changes alter the loss distribution, and the lower-rate answer variance is unmeasured. The net-accuracy threshold also differs from a one-directional loss target. This scaling cannot establish power for either low-rate outcome.

D.5 Concentration of losses

For each reader and workload, we use random-link comparisons at $d = .8$ in which the correct-index answer is correct and the context changes. Only these comparisons can lose an answer in the saved data, because identical requests reuse their

Table 13: Conditional normal-approximation planning for gpt-oss random links at $d = .8$. The threshold is 2.8 times the estimated standard error. Question totals use inverse-square scaling. Workplace calculations assume independent questions within five fixed histories; social calculations require new independent histories with comparable composition. These are planning scenarios, not validated population power.

Workload	Threshold (points)	For 2 points	For 1 point
Workplace	2.96	440	1757
Social	5.83	679	2716

response. As a reference, we assign a loss independently to each eligible comparison with one common probability, the pooled unweighted loss rate. We retain the original question weights when computing concentration. We use 5,000 simulations for the share in the most affected 10% of questions and 20,000 for the number of questions with any loss and the top-question share. Table 14 compares the observed measures with this reference. For b simulated values at least as concentrated as the observation and B simulations with a defined measure, the one-sided simulation tail fraction is $(b + 1)/(B + 1)$. Zero-loss simulations count toward the number-of-questions comparison but are excluded from the share measures, whose denominator is then zero. The top-10% simulation uses seed 20261014; the count and top-question simulations use seed 7. Each seed restarts for each reader and workload. Separate fixed streams compute the top-10% share and the count/top-question statistics. We report all three statistics for each cohort. These post hoc comparisons are unadjusted for multiple tests.

The simulation treats comparison rows as independent, although they can share a baseline response, a changed request, or a history. The four option orders for one question and draw also share the same context. It is therefore a descriptive model reference, not a calibrated test against every form of random variation. Departures can reflect these dependencies or unequal question risk. They do not identify why losses concentrate or establish vulnerability specific to false links.

E Full result tables

Table 15 reports order ranges from unrounded values. Hosted-reader transitions can include service variation; no summary subtracts repeat-request disagreement. Table 16 partitions the answer changes, and Table 17 separates changed contexts by label-set membership. Table 18 omits the selected Marcus history. Table 19 groups answers by no-memory and full-evidence controls.

Message overlap uses Jaccard similarity $J = |M_c \cap M_a|/|M_c \cup M_a|$, with empty/empty similarity defined as one. It describes overlap of selected sets, not the fraction of messages removed from the correct-index context. The latter fraction is $|M_c \setminus M_a|/|M_c|$ when M_c is nonempty. Evidence partitions and overlap bins are descriptive associations with answer changes. Table 20 gives the matched full-evidence contrasts.

Table 21 compares valid answers with the same-order no-memory answer. Stable answers can be correct or incorrect; agreement alone does not identify how memory was used.

Table 22 reports identical-request disagreement.

Repeat-sample size and composition prevent estimating a noise contribution to each false-link contrast. Qwen's zero disagreements remain a finite-sample result.

F Packing condition and proof

The packing rule compares messages present on only one side with space left by the shared messages. Let S_0, S_1 be shortlists, $C = S_0 \cap S_1$ their common set, and $E = (S_0 \cup S_1) \setminus C$ their exclusive messages. Let $P(S)$ be scan-once packing under budget $B \geq 0$. Message costs $\ell(t)$ are fixed and positive, order $\prec$ is fixed across sets, and exact fits are admitted. Before an exclusive message h, packing only the common messages leaves

$$R_C(h) = B - \sum_{t \in P(C):\, t \prec h} \ell(t).$$

The necessary and sufficient condition is

$$\begin{aligned} P(S_0) = P(S_1) \quad &\iff \\ \ell(h) > R_C(h) \quad &\text{for every } h \in E. \end{aligned} \tag{3}$$

For sufficiency, insert the exclusive messages in total order into the common-set execution. Each is skipped because its cost exceeds the unchanged residual. Every common message therefore keeps the same packing decision. For necessity, equal

Table 14: Loss concentration under random links at $d = .8$, against an independent equal-risk reference. Each cell gives observed / simulation median / simulation tail fraction. The reference assigns each eligible comparison the pooled loss probability and retains question weights. It does not preserve shared-request or history dependence. For gpt-oss social, only 5 questions have any loss, fewer than the eight top-10% slots. The observed share is therefore necessarily 100%; the count and top-question measures resolve this ceiling.

Reader	Workload	Questions with any loss	Top 10% share (%)	Top question share (%)
gpt-oss	Workplace	75 / 99 / < 0.001	48 / 38 / < 0.001	4 / 3 / 0.005
gpt-oss	Social	5 / 15 / < 0.001	100 / 71 / 0.020	79 / 15 / < 0.001
GPT-4.1 mini	Workplace	14 / 16 / 0.313	68 / 58 / 0.169	16 / 11 / 0.116
GPT-4.1 mini	Social	11 / 13 / 0.269	87 / 74 / 0.159	28 / 15 / 0.035

Table 15: Principal gpt-oss accuracy contrasts in percentage points. Link conditions average three draws and all conditions average four orders. Workplace ranges omit each history in turn; social 95% intervals resample histories. Order ranges show the smallest and largest contrast across the four cyclic orders.

Contrast	Workplace			Social		
	Estimate	Omit-one range	Order range	Estimate	95% interval	Order range
Correct index − unfiltered search	-0.63	[-3.12, +0.94]	[-4.00, +5.00]	+0.51	[-3.86, +6.61]	[-1.22, +2.44]
Random links − correct index	-1.75	[-2.71, -0.52]	[-3.33, +0.33]	-1.52	[-6.10, +1.32]	[-1.90, -1.22]
Look-alike links − correct index	-0.04	[-1.04, +1.20]	[-1.33, +1.17]	-1.46	[-5.42, +1.12]	[-2.30, -0.54]
Look-alike links − random links	+1.71	[+1.04, +2.71]	[+0.83, +2.00]	+0.07	[-1.32, +1.46]	[-0.54, +0.68]

Table 16: Complete hosted-reader answer transitions (%). All conditions are compared with the correct index. Gain, loss, valid wrong-to-wrong change, and invalid wrong change sum to churn before rounding. Net is gain minus loss in percentage points. All rows average four orders; links average three draws for gpt-oss and one for GPT-4.1 mini.

Reader	Workload	Condition	Churn	Gain	Loss	Wrong	Invalid	Net
gpt-oss	Workplace	Unfiltered search	27.88	10.75	10.13	7.00	0.00	+0.63
gpt-oss	Workplace	Look-alike links	20.04	7.46	7.50	5.08	0.00	-0.04
gpt-oss	Workplace	Random links	19.25	6.04	7.79	5.38	0.04	-1.75
gpt-oss	Social	Unfiltered search	5.18	2.34	2.85	0.00	0.00	-0.51
gpt-oss	Social	Look-alike links	4.17	1.08	2.54	0.54	0.00	-1.46
gpt-oss	Social	Random links	4.07	1.05	2.57	0.44	0.00	-1.52
GPT-4.1 mini	Workplace	Unfiltered search	15.94	5.94	4.69	5.31	0.00	+1.25
GPT-4.1 mini	Workplace	Look-alike links	8.75	2.81	3.44	2.50	0.00	-0.62
GPT-4.1 mini	Workplace	Random links	15.62	6.56	5.94	3.12	0.00	+0.62
GPT-4.1 mini	Social	Unfiltered search	10.01	3.00	4.07	2.95	0.00	-1.07
GPT-4.1 mini	Social	Look-alike links	5.95	1.22	3.20	1.52	0.00	-1.98
GPT-4.1 mini	Social	Random links	7.52	1.68	4.32	1.52	0.00	-2.64

selected sets cannot contain an exclusive message. Removing these skipped messages leaves every residual and common decision unchanged, reducing both executions to $P(C)$. Each removed message must thus exceed its common-set residual.

The earliest exclusive message with $\ell(h) \leq R_C(h)$ identifies the first unequal selection. Earlier exclusive messages are skipped, so this message's side follows the common execution until admitting it. The rule handles both additions and removals caused by the shortlist cap. Figure 5 gives a small example.

Under the study's fixed serialization, equal selected sets give identical context. In a system where distinct identifiers can serialize to the same text, byte equality requires a separate check. Set-dependent scores, listwise reranking, and generated summaries violate the fixed-order or fixed-cost assumptions and need a different analysis.

G Qwen results and option-order checks

Qwen's primary endpoint compares the correct index with unfiltered search in the released option order. Both social conditions score 55.5%, with a paired interval of $[-9.1, 8.1]$ points for their difference. The workplace difference is -1.5 points. Table 23 retains this negative endpoint and the equal-question sensitivity. Equal-history weighting gives a question from a one-question history more mass than a question from a larger history.

Table 17: Random-link comparisons at $d = .8$ with changed context, split by evidence-label membership. Pairs are unweighted counts. Churn, gain, and loss are conditional percentages under original history mass. The social percentages therefore differ from pooled pair fractions.

Reader	Workload	Labels	Pairs	Churn	Gain	Loss
gpt-oss	Workplace	unchanged	1396	22.78	7.02	9.24
gpt-oss	Workplace	changed	548	26.28	8.58	10.58
gpt-oss	Social	unchanged	708	2.90	1.50	0.77
gpt-oss	Social	changed	20	62.50	0.00	62.50
GPT-4.1 mini	Workplace	unchanged	188	14.89	5.32	4.79
GPT-4.1 mini	Workplace	changed	88	25.00	12.50	11.36
GPT-4.1 mini	Social	unchanged	244	7.91	2.33	4.31
GPT-4.1 mini	Social	changed	4	75.00	0.00	50.00

Table 18: Effect after omitting the Marcus history (percentage points). The remaining 40 social histories receive equal weight. This post hoc influence check uses a fixed-seed history bootstrap conditional on saved responses. Its intervals do not account for selecting Marcus after inspecting losses.

Reader	Links	Without Marcus	95% interval
gpt-oss	Look-alike links	+0.17	[-1.08, +1.32]
gpt-oss	Random links	+0.52	[-0.17, +1.60]
GPT-4.1 mini	Look-alike links	-2.03	[-4.22, -0.10]
GPT-4.1 mini	Random links	-1.46	[-3.96, +0.78]

Table 19: Random-link effect on question-order cells classified by control answers. Dependent means no memory is wrong and full evidence is right. Other means the matched full-evidence cohort outside that class. Unavailable means a full-evidence response is absent. Counts are distinct question-order cells, before draw expansion; mass retains original weights. Effects are percentage points.

Reader	Workload	Stratum	Cells	Mass	Effect
gpt-oss	Workplace	full evidence available	784	98.00	-1.91
gpt-oss	Workplace	full evidence unavailable	16	2.00	+6.25
gpt-oss	Workplace	memory dependent	313	39.13	-4.05
gpt-oss	Workplace	other matched cells	471	58.88	-0.50
gpt-oss	Social	full evidence available	320	100.00	-1.52
gpt-oss	Social	full evidence unavailable	0	0.00	–
gpt-oss	Social	memory dependent	214	64.63	-1.52
gpt-oss	Social	other matched cells	106	35.37	-1.53
GPT-4.1 mini	Workplace	full evidence available	316	98.75	+0.63
GPT-4.1 mini	Workplace	full evidence unavailable	4	1.25	+0.00
GPT-4.1 mini	Workplace	memory dependent	148	46.25	+4.73
GPT-4.1 mini	Workplace	other matched cells	168	52.50	-2.98
GPT-4.1 mini	Social	full evidence available	320	100.00	-2.64
GPT-4.1 mini	Social	full evidence unavailable	0	0.00	–
GPT-4.1 mini	Social	memory dependent	200	61.69	-3.38
GPT-4.1 mini	Social	other matched cells	120	38.31	-1.46

Four-order averaging changes the workplace contrast's sign (Table 24). Both columns use the first random draw at $d = .8$, unlike Qwen's primary three-draw analysis. Ordinary conditions use the full cohort; full-evidence comparisons use the common feasible subset (Appendix C).

The 32 workplace update questions ask how revised information changes a decision. Qwen answers none correctly in the released order under no memory, unfiltered search, the correct index, or any of the three $d = .8$ draws. Consequently, its conditional loss risk on these questions is undefined, since no correct-index answer can be lost. The floor changes under option reordering. Only 28 update questions fit the full-evidence control in every order. All update questions remain in the primary endpoint.

Among changed-context pairs with stable answers at $d = .8$, Qwen matches its no-memory answer on 86.2% of workplace and 29.7% of social comparison mass in the released order. Table 21 gives the separate four-order, one-draw check. Complete-quartet identity consistency requires four valid responses; pairwise disagreement

Table 20: Full-evidence minus comparator accuracy in percentage points. Each comparison retains identical questions in both conditions and all four option orders. Workplace ranges omit histories; social ranges are 95% history-bootstrap intervals. Full evidence uses all labeled messages outside the fixed memory budget, so it changes both evidence availability and context length.

Reader	Workload	Comparator	Questions	Δ	Range
gpt-oss	Workplace	Correct index	196	+20.90	[+18.87, +21.91]
gpt-oss	Workplace	Unfiltered search	196	+20.35	[+18.39, +22.16]
gpt-oss	Workplace	no memory	196	+34.48	[+33.88, +35.44]
gpt-oss	Social	Correct index	80	+16.06	[+6.40, +27.44]
gpt-oss	Social	Unfiltered search	80	+16.57	[+6.61, +28.05]
gpt-oss	Social	no memory	80	+63.82	[+55.28, +71.34]
GPT-4.1 mini	Workplace	Correct index	79	+29.00	[+25.70, +31.56]
GPT-4.1 mini	Workplace	Unfiltered search	79	+27.73	[+25.68, +30.08]
GPT-4.1 mini	Workplace	no memory	79	+44.56	[+40.47, +47.50]
GPT-4.1 mini	Social	Correct index	80	+21.24	[+8.13, +33.74]
GPT-4.1 mini	Social	Unfiltered search	80	+22.31	[+8.94, +35.37]
GPT-4.1 mini	Social	no memory	80	+58.94	[+47.46, +68.90]

Table 21: Conditional agreement with the same-order no-memory answer (%). Stable/context changed requires changed context and the same valid answer. New answer requires two different valid answers and compares the changed condition's answer with no memory. Rates retain original comparison weights. gpt-oss uses three draws; Qwen and GPT-4.1 mini use one. All rows average four orders. GPT-4.1 mini has 80 workplace questions; the other readers have 200.

Reader	Workload	Links	Stable/context changed	New answer
Qwen	Workplace	Random links	82.32	42.42
Qwen	Social	Random links	33.45	34.10
gpt-oss	Workplace	Random links	63.90	40.35
gpt-oss	Workplace	Look-alike links	64.60	38.25
gpt-oss	Social	Random links	29.74	5.83
gpt-oss	Social	Look-alike links	31.33	14.63
GPT-4.1 mini	Workplace	Random links	72.57	32.00
GPT-4.1 mini	Workplace	Look-alike links	75.00	60.71
GPT-4.1 mini	Social	Random links	30.24	32.43
GPT-4.1 mini	Social	Look-alike links	34.16	32.48

Table 22: Different answers to identical requests, as count/total and exact 95% binomial intervals (%). Requests were selected within fixed condition strata. Intervals assume a common independent Bernoulli process within each listed sample; selection limits their interpretation beyond those samples. The reproduction files retain all condition strata.

Reader	Workload	Changed / pairs	Exact 95% interval
Qwen	All	0/20	[0.00, 16.84]
gpt-oss	All	8/48	[7.48, 30.22]
gpt-oss	Workplace	3/24	[2.66, 32.36]
gpt-oss	Social	5/24	[7.13, 42.15]
GPT-4.1 mini	All	1/24	[0.11, 21.12]

uses only valid response pairs.

H Case studies

The Marcus trace expands Figure 1. Figure 6 shows the contrasting Camila case, where the direct supporting statement stays in the context.

H.1 Marcus and the rules before game night

The question asks who gave the more specific procedure for reading game rules before a session. Released options are Tomasz, Derek, Marcus, and Raj; Marcus is correct. Tomasz proposes choosing the game early so people can read the rules. Marcus adds a Wednesday deadline for the owner's rulebook link. His statement is the sole evidence-labeled message.

With the correct index, gpt-oss selects Marcus in all four option orders. Random links at $d = .8$ remove his labeled message from all 12 compared contexts. The reader selects Tomasz ten times

Table 23: Qwen correct-index minus unfiltered-search accuracy in the released option order. Differences are percentage points. The range uses workplace omit-one-history means or a social 95% paired history-bootstrap interval. Gain and loss columns are unweighted question counts.

Workload	Equal hist. pp	Question pp	Range	$0 \to 1$	$1 \to 0$
Workplace	-1.5	-1.50	[-3.1, +0.0]	5	8
Social	+0.0	+3.75	[-9.1, +8.1]	7	4

Table 24: Qwen option-order sensitivity in percentage points. The first column uses the released order; the second averages four cyclic orders. Random links use one fixed draw at $d = .8$ in both columns. Ranges use workplace omit-one-history means or social history-bootstrap intervals.

Contrast	$r = 0$ (pp)	Four orders (pp)
Workplace		
Correct index − Unfiltered search	-1.5 [-3.1, +0.0]	+1.2 [+0.8, +1.7]
Random links − Correct index	+3.5 [+1.9, +5.0]	-0.5 [-1.1, +0.0]
Social		
Correct index − Unfiltered search	+0.0 [-9.1, +8.1]	-3.9 [-9.4, +1.2]
Random links − Correct index	-2.6 [-9.3, +4.9]	+1.3 [-2.4, +5.7]

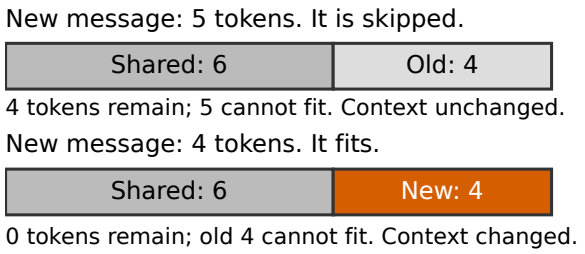


Figure 5: An exclusive message changes selected membership only if it fits. In this constructed example, the first shared message leaves four tokens. An exclusive five-token message is skipped. An exclusive four-token message fits and displaces the later shared message.

and Marcus twice. Draws one and three give four Tomasz answers each; draw two gives two Tomasz and two Marcus answers. No-memory answering is correct in one of four orders. Full evidence is correct in all four. Unfiltered search omits the labeled message and selects Tomasz in all four orders.

The first random draw has five false links. They connect Marcus with Derek, Priya with Tomasz, Priya with Derek, Raj with Felix, and Nadia with Soo-Yeon. The resulting context admits Derek's messages "agree. read the rules people" and "read it. lets go", plus Soo-Yeon's comment that Felix did well for a first game. Marcus's rulebook instruction and one Priya message leave. Figure 1 shows selected messages from this comparison. All displayed names and message texts stay correct.

In draw two, three Lena messages and one Kezia message enter. Marcus's instruction, two Priya messages, and one Tomasz message leave. In draw three, two Derek messages, one Kezia message, and one Soo-Yeon message enter, while Marcus's instruction and two Priya messages leave. The reproduction files retain all context messages, source identifiers, answers, and graphs for this trace.

The first look-alike draw retains Marcus's message and yields four correct answers. The other two omit it and yield eight Tomasz answers. Across random links, look-alike links, and unfiltered search, Tomasz is selected in 22 of the 24 comparisons where that message is absent. The question was selected after decomposing social losses. As its history's only evaluated question, it has weight $1/41$. Multiple messages change together, and hosted service variation remains possible, so the trace does not isolate one message's causal effect.

Qwen gives a contrasting result on the same question at $d = .2$ in the released order. Marcus's instruction remains in both contexts. Two turns enter and two leave, and Qwen changes from Tomasz to Marcus. This gain in key agreement occurs without recovering the labeled support.

H.2 Camila and support outside the evidence label

The Camila question asks who preferred good-enough observing conditions every night to perfect conditions a few times a year. The options are Bex,

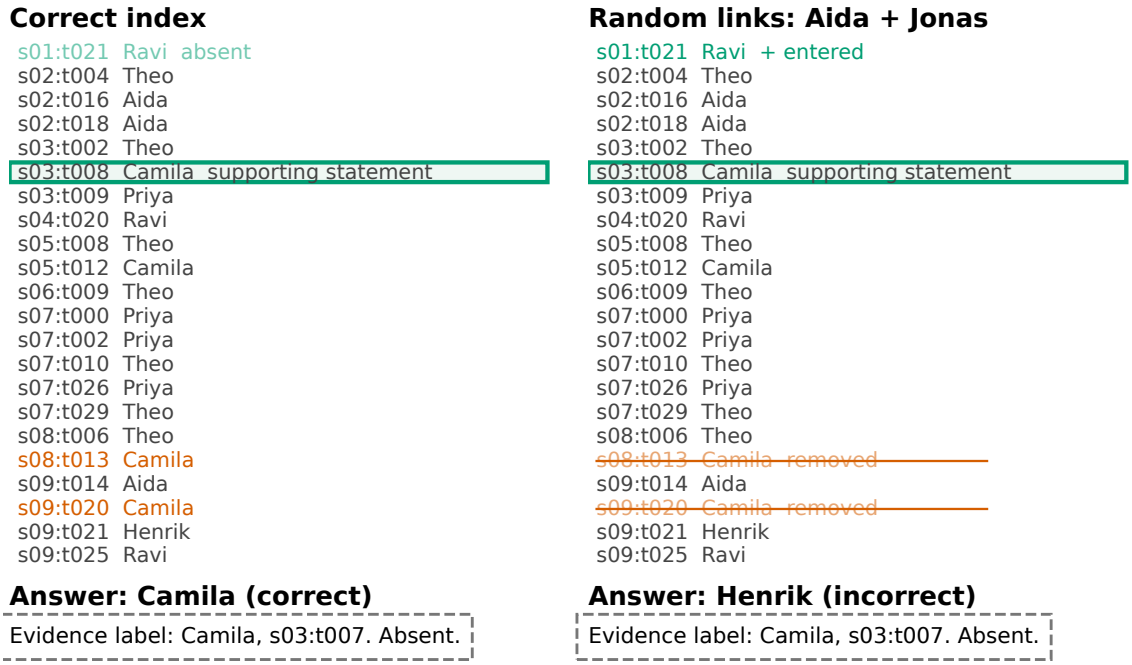


Figure 6: Qwen changes its answer although Camila's supporting statement remains. Jonas's messages reach the shortlist but do not fit the context. One Ravi message enters and two Camila messages leave. The benchmark's labeled turn is absent from both contexts. Each column lists the union of selected message identifiers, with absent messages marked.

Camila, Henrik, and Marcus. In the first random draw at $d = .2$, an Aida–Jonas link admits Jonas's messages. Two reach the shortlist, but neither enters the final context. The changed packing instead admits one Ravi message and removes two Camila messages.

Camila's correctly named statement remains in both contexts, but Qwen changes from Camila to Henrik. The benchmark labels an adjacent earlier Camila turn, which is absent in both contexts. Consequently, both contexts have zero labeled coverage even though the direct supporting statement remains. Figure 6 shows the complete union of selected message identifiers and marks absence on each side.

This case comes from a fixed pseudorandom selection of one gain, loss, stable-correct, and stable-wrong case per workload and contrast, for two contrasts. All 16 cases, contexts, and responses remain in the reproduction files. Selection supports outcome inspection, not frequency estimation.

I Limits of the evidence labels

Workplace questions have a median of 26 evidence labels; social questions have one. Workplace labels often include background, while labels can omit direct support, as for Camila. Coverage therefore has different meanings across workloads.

An AI assistant inspected 30 questions, one per eligible family and workplace history. Fixed pseudorandom selection followed retrieval analysis but preceded reader-answer inspection. Complete labeled spans, options, and keys were visible. Some questions depended on style; some labels omitted the requested detail. These judgments are neither human annotations nor an error-rate estimate. Records retain unresolved cases, including two update relations. Scoring retains official keys.

Full evidence exceeds the retrieval budget and can include background or superseded statements. It is a context control on the feasible cohort. Workplace profile questions can require history-wide patterns and are excluded from the benchmark's oracle evaluation (Hu et al., 2026). All conditions use complete histories because questions have no time cutoff.