RELEVANCE IS NOT SUFFICIENCY: WHAT ACTUALLY CLOSES THE EVIDENCE GAP IN LONG-TERM MEMORY QA

Anonymous authors
Paper under double-blind review

ABSTRACT

LLM agents that interact with a user across many sessions accumulate histories that exceed their context window, so they store past interactions in an external memory and answer each question from a small set of retrieved records. Existing memory systems rank records by lexical or embedding relevance, yet the top-ranked memories can each be relevant while jointly omitting a complementary fact that the answer requires, especially for multi-session and temporal questions. Drawing on the distinction between relevance and sufficiency in legal evidence scholarship, we recast memory retrieval as constructing a *sufficient memory set*. To operationalize this view, we introduce a blinded LLM judgment over the retrieved set, together with Gold Hit and Turn Hit as evidence-coverage proxies. We then propose *Budgeted Flat Reconstruction (BFR)*, which builds sufficient sets over a fixed flat memory store in two stages. Specifically, we first apply *Formal Concept Analysis for Memory Selection (FCA-MS)* to decompose the question into information requirements and select a compact candidate subset that jointly covers them. Then, we repeatedly acquire unseen records through deeper text search or complementary entity and session views, stopping when the budget is exhausted. Experiments on LoCoMo and LongMemEval-S show that BFR outperforms same-store adaptations of recent agent-memory systems in both answer quality and evidence coverage. Specifically, on LongMemEval-S it raises judged accuracy from 72.4% to 82.2% and Turn Hit to 91.4%.

1 INTRODUCTION

LLM agents interact with users across many sessions, accumulating preferences, events, and other information that may matter much later (Zhong et al., 2024; Maharana et al., 2024; Wu et al., 2025; Tan et al., 2025). Because the full interaction history is too long to remain in the context window (Packer et al., 2023), these agents commonly rely on external memory to retain and access past information. Agent memory has therefore become an important component for maintaining continuity across conversations and answering questions about earlier interactions. Recent agent-memory systems improve answer quality by designing how memories are represented and retrieved. For example, Mem0 distills conversations into salient facts, consolidates them over time, and retrieves them by embedding similarity (Chhikara et al., 2025). A-Mem instead maintains evolving, interlinked notes retrieved in the same way (Xu et al., 2025). CoM builds coherent memory chains from relevance-ranked candidates (Xu et al., 2026), while MRAgent reasons over a Cue–Tag–Content graph (Ji et al., 2026; Hu et al., 2025; Luo et al., 2026). Despite their different memory structures, these systems typically acquire candidates by relevance, whether through lexical matching, embedding similarity, or guided traversal (Zhang et al., 2025). Yet relevance is a property of each record, not of the set, i.e., a ranking can return several memories about the right topic while the one partner fact the answer requires falls below the cutoff. It means that **retrieving relevant memories does not by itself ensure that the resulting set can support the answer.** This raises a question: what determines whether the retrieved records made available to an agent support a correct answer?

Legal evidence scholarship offers a useful distinction: *relevance* concerns whether an individual item bears on a claim, whereas *sufficiency* concerns whether the evidence considered together supports a finding (Lempert, 1986; Pardo, 2023). We apply this distinction to memory QA and ask

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

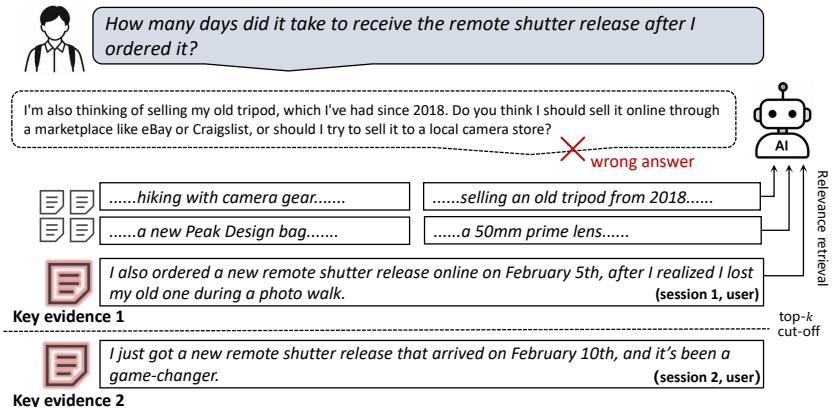


Figure 1: **A real error: relevant but insufficient records.** The question asks how many days the remote shutter release took to arrive. Relevance ranking returns Key evidence 1 together with other photography memories, while Key evidence 2 falls below the top-k cutoff.

whether the retrieved memories, taken together, contain enough information to determine the answer. Figure 1 illustrates this failure with a real LONGMEMEVAL-S case. The question asks how many days a remote shutter release took to arrive, which requires two records together: the February 5 order and the February 10 arrival. The top-k retrieved memories include the order alongside other photography-related records, but the arrival, a less similar record from another session, falls below the cutoff. The agent then responds about selling an old tripod instead of the five-day interval. This example shows that individually relevant records do not guarantee a sufficient set.

We therefore recast memory retrieval as the construction of a *sufficient memory set*, one from which the answer can be determined. Constructing such a set raises two coupled challenges. *First, a set must bring together complementary facts.* A question may depend on several pieces of information, and a limited context budget can be spent on records that repeat one piece while leaving another uncovered. *Second, the necessary facts must be reachable.* Evidence missing from the initial candidate pool cannot be recovered by improving selection within that pool. Both challenges must be met without knowing which records are required. At inference time, no oracle declares a set sufficient, so a method must itself decide how far to search and when to stop.

To address both challenges, we propose *Budgeted Flat Reconstruction* (**BFR**), a two-stage, requirement-aware framework that assembles answer-supporting evidence over a flat memory store under a fixed retrieval budget. Specifically, at Stage I, *Formal Concept Analysis for Memory Selection* (FCA-MS) first decomposes the question into answer requirements, and then applies an FCA-based prefilter followed by LLM selection of a compact, complementary subset. These selected candidates are grounded to source records to form the initial evidence set. At Stage II, we acquire further supporting evidence from the current evidence set, excluding records already obtained and accumulating new support within a fixed retrieval budget. We instantiate this stage with two schedules: *Budgeted Flat Reconstruction with Text Access* (BFR-Text) extends text retrieval, and *Budgeted Flat Reconstruction with Multi-View Access* (BFR-MV) combines text, entity, and session views. Stage I thus composes the evidence already present among the candidates, Stage II reaches evidence beyond them, and the fixed budget supplies a stopping rule that does not presume sufficiency can be recognized. Finally, we evaluate BFR on LOCOMO and LONGMEMEVAL-S under protocols that hold the memory store, answerer, and judge fixed within each benchmark. Extensive results show that BFR improves judged answer quality and annotated evidence coverage over same-store baselines. On LONGMEMEVAL-S, it raises judged accuracy from 72.4% to 82.2% and attains a 91.4% Turn Hit. In summary, this paper makes three contributions:

(i) We formulate memory retrieval as the construction of a sufficient memory set and make set-level sufficiency measurable: LLM-Suff@Set judges whether a retrieved set supports the answer, while Gold Hit and Turn Hit track coverage of annotated answer-bearing evidence.

(ii) We propose BFR, a two-stage method for building sufficient sets over a flat memory. Its two stages recover complementary evidence missing within and beyond the initial candidates.

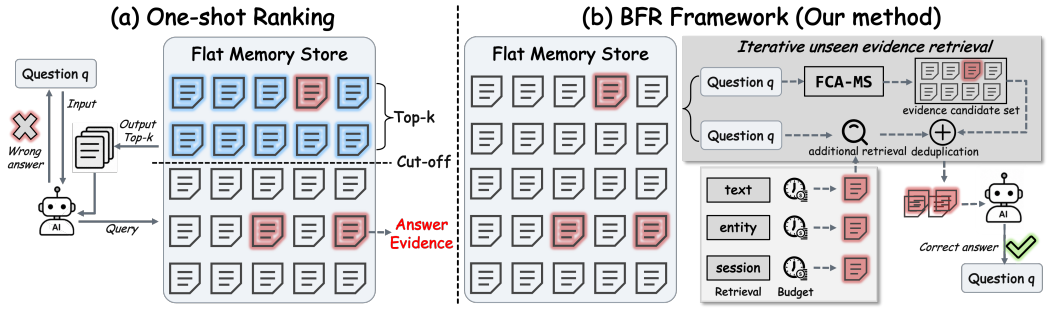


Figure 2: **Similarity top-k vs. set-level sufficiency.** (a) Prior memory architectures rank records by query similarity and return the top-k slice. (b) BFR returns a sufficient set over the flat store: it first selects complementary records that jointly cover the question's requirements, then completes missing support under a budget by adding unseen records through text, entity, and session views.

(iii) We conduct extensive experiments on LoCoMo and LongMemEval-S. These results show that BFR improves answer quality and evidence coverage, and identify where additional evidence is recovered and clarify the distinct roles of evidence composition and acquisition.

2 Related Work

Evidence relevance and sufficiency. Evidence scholarship treats relevance as a property of an item's bearing on a claim and sufficiency as a judgment over the complete evidentiary record (Pardo, 2023). In RAG, sufficient-context evaluation uses this distinction diagnostically, separating failures caused by missing context from failures to use available context (Joren et al., 2025). We bring this distinction to long-term agent memory and make it constructive: rather than diagnosing whether a retrieved set was sufficient, retrieval should assemble and complete the evidence set that supports an answer. Appendix A spells out the mapping from Pardo's record-level test to our method.

Memory construction and relevance-based access. Memory systems increasingly transform interaction history before retrieval. Mem0 maintains salient facts through add, update, and delete operations and retrieves semantically similar memories (Chhikara et al., 2025). A-Mem creates evolving linked notes but still answers from a top-k similarity-ranked subset (Xu et al., 2025). Zep represents temporal validity in a bi-temporal graph (Rasmussen et al., 2025); MemoryOS uses hierarchical storage (Kang et al., 2025); and SeCom segments conversations into coherent units (Pan et al., 2025). CoM improves post-retrieval organization by growing chains from top-ranked anchors according to query relevance and contextual consistency (Xu et al., 2026). MRAgent applies active reconstruction on a Cue–Tag–Content memory graph, allowing an LLM to choose retrieval tools from accumulated evidence (Ji et al., 2026). These systems improve what can be retrieved or how relevant candidates are organized. BFR instead asks whether the resulting set covers all answer requirements, then acquires missing support without changing the underlying store.

3 What Counts as a Sufficient Candidate Memory Set

Motivation. Similarity search can return several memories about the right topic while missing the fact that makes an answer possible. In figure 1, the retrieved records include the shutter-release order and other photography-related memories, but not the arrival date. The records are relevant to the question; together they cannot establish the delivery interval. Legal evidence scholarship makes the same item-versus-set distinction: an item can bear on a claim without the assembled record being sufficient to support a finding (Lempert, 1986; Pardo, 2023) (Appendix A). For memory QA, the practical difficulty is twofold. Relevant candidates may repeat one part of the answer instead of supplying complementary facts, and a necessary fact may fall outside the initial candidate pool even when it remains in the memory store. Figure 2 illustrates why a memory system must address both the composition of the retrieved set and access to evidence beyond the first retrieval boundary.

Quantifying candidate-set sufficiency. Let q be a question, $y^\star(q)$ its reference answer, $\mathcal{M}$ a fixed memory store, and $E \subseteq \mathcal{M}$ the retrieved candidate set. We write $\mathrm{text}(E)$ for the evidence text

Table 1: **Illustrative delivery-time question.** Both dates are required to derive the five-day answer.

Retrieved evidence	LLM-Suff@Set	Gold Hit	Turn Hit
Order only (Feb. 5)	0	1	0
Arrival only (Feb. 10)	0	1	1
Order + arrival	**1**	1	1

rendered from E and $\mathrm{src}(E)$ for the set of source-record or source-turn identifiers linked to E. As usual, $\mathbb{1}[P]$ equals 1 when proposition P holds and 0 otherwise. The ideal set-level property is

$$S(q, E) = \mathbb{1}[y^\star(q) \text{ is determinable from } q \text{ and } \mathrm{text}(E)]\,. \tag{1}$$

Thus, $S(q, E)$ concerns what the evidence makes answerable, independently of whether a particular answerer generates the correct response. To assess this property offline, let $a(q, E)$ be the final *decided* label from a blinded, reference-conditioned LLM adjudication given q, $y^\star(q)$, and $\mathrm{text}(E)$ (Appendix M), with $a(q, E) \in \{\text{sufficient}, \text{insufficient}\}$. We define ***LLM-Suff@Set*** by

$$\widehat{S}_{\mathrm{LLM}}(q, E) = \mathbb{1}[a(q, E) = \text{sufficient}]\,. \tag{2}$$

Here, $\widehat{S}_{\mathrm{LLM}}(q, E)$ is the operational, reference-conditioned assessment of joint answer support. It is undefined when adjudication is uncertain or missing; such cases are excluded from the reported rate rather than counted as insufficient. For questions with annotated supporting records, let $\mathcal{G}(q)$ be their nonempty set of source identifiers and $\mathcal{H}(q)$ be their set of annotated source-turn identifiers, which may be empty. *Gold Hit* and *Turn Hit* are defined as follows:

$$G(q, E) = \mathbb{1}[\mathrm{src}(E) \cap \mathcal{G}(q) \neq \varnothing] \tag{3}$$

$$\mathrm{TH}(q, E) = \mathbb{1}[\mathrm{src}(E) \cap \mathcal{H}(q) \neq \varnothing] \tag{4}$$

The value $G(q, E)$ is 1 if E reaches at least one annotated supporting record; it does not require retrieval of every supporting record. Gold Hit is undefined when $\mathcal{G}(q) = \varnothing$ and is averaged over questions with nonempty supporting-record annotations. The value $\mathrm{TH}(q, E)$ is 1 if E reaches at least one annotated answer-bearing turn. Our main LONGMEMEVAL-S results average Turn Hit over all 500 questions. These proxies measure whether annotated evidence was reached, not whether the set suffices. In Table 1, each singleton has Gold Hit 1, and the arrival alone has Turn Hit 1. Neither yields the five-day answer, but together they do. A hit is therefore necessary for sufficiency but does not establish it, which is why we report the proxies alongside, not in place of, the set-level judgment.

4 BFR: COMPOSING AND COMPLETING SUFFICIENT EVIDENCE

4.1 PROBLEM: CONSTRUCT A SUFFICIENT SET UNDER A RETRIEVAL BUDGET

Let $\mathcal{M} = \{m_1, \ldots, m_N\}$ be a fixed memory store, q a question, and $y^\star$ its reference answer. An access policy constructs $E \subseteq \mathcal{M}$, after which the fixed answerer f emits $\hat{y} = f(q, E)$. The set-level sufficiency property $S(q, E)$ is defined in Section 3. Holding $\mathcal{M}$ and f fixed, the construction problem is to maximize expected sufficiency under a retrieval-call budget B:

$$\max_{\pi}\ \mathbb{E}[S(q, E_T)] \quad \text{s.t.} \quad C_{\mathrm{ret}}(\pi, q) \leq B. \tag{5}$$

A one-shot reader returns $E = \mathrm{TopK}(q, \mathcal{M})$ and cannot recover once a required exhibit is missing. Budgeted reconstruction instead starts from an initial set E_0 and adds only unseen records,

$$E_{t+1} = E_t \cup K_t, \qquad K_t \subseteq \mathcal{M} \setminus E_t. \tag{6}$$

Equation 5 is the problem. The two gaps in Section 3 say what a solution must do. Composition chooses E_0 so that distinct answer requirements are covered rather than redundant relevant items. Completion chooses each K_t so that a missing exhibit can still enter the record after the first page.

BFR assigns one stage to each gap. Let $\widetilde{\mathcal{M}}$ be an agent-memory note representation and ϕ a grounding map into $\mathcal{M}$. Stage I builds E_0 with FCA-MS and Stage II runs a completion schedule π:

$$\begin{aligned} E_0 &= \phi\Big(\rho_{\mathrm{FCA}}(q, \widetilde{\mathcal{M}})\Big)\,, \\ A_t &= \pi(q, E_t, t), \\ E_{t+1} &= E_t \cup \kappa_\pi(q, E_t, C_t), \qquad t < T. \end{aligned} \tag{7}$$

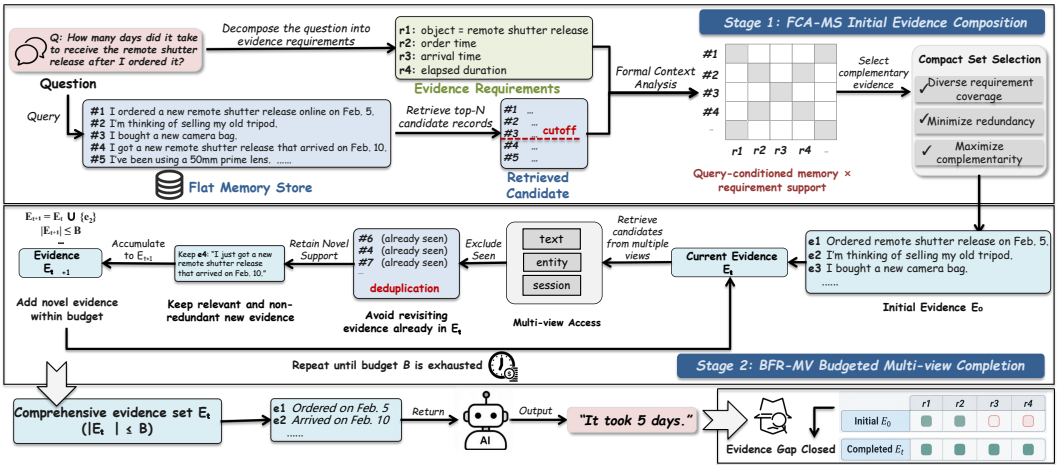


Figure 3: **The two-stage evidence-construction pipeline in BFR.** Stage I (FCA-MS) retrieves a candidate pool larger than the usual top-k, decomposes the query into answer requirements, prefilters candidates with FCA, and has an LLM select a compact subset that jointly covers the requirements. The selected candidates are mapped to source records to form E_0. Stage II (shown for BFR-MV instantiation) completes the set under a fixed budget: text, entity, and session access retrieve unseen records. Only new records are accumulated into E_{t+1} until the budget is exhausted. The comprehensive evidence set returns to Agent, and Agent answers the question based on the set.

A_t names the retrieval views and cues at round t, C_t is the unseen candidate pool those calls return, and κ_π retains added records. The answerer reads E_T after the budget is spent. Figure 3 makes this data flow explicit. Stage I asks the agent memory system for an expanded candidate pool beyond its usual top-k, decomposes the query into requirements, filters the pool by requirement-aware FCA, and asks an LLM to select a compact supporting subset. A bounded grounding map then supplies the initial source-record evidence E_0. Stage II treats this set as a cumulative evidence state, retrieves through the views specified by π, removes records already in E_t, and retains novel support until an observable budget is exhausted. The figure instantiates this loop with BFR-MV; BFR-Text uses the same exclusion, retention, accumulation, and stopping operations with text access only. Appendix A records how S, R, $\mathcal{R}_q$, and the two stages correspond to Pardo's item/record distinction.

4.2 Stage I: Compose Complementary Evidence with FCA-MS

Expanded retrieval and requirement decomposition. An agent memory system returns an ordered pool $\mathcal{C}_q = \mathrm{TopN}(\mathrm{Retrieve}_{\mathrm{agent}}(q, \widetilde{\mathcal{M}}))$, with $N > k$ relative to its usual top-k retrieval cutoff. An LLM decomposes q into answer-oriented requirements $\mathcal{R}_q = \{r_j\}_{j=1}^{J}$, including distinct facts, bridges, list items, and temporal constraints.

Requirement-aware prefilter. An FCA-based prefilter uses the requirements and retrieved candidates to retain a smaller pool $\mathcal{P}_q = \mathrm{FCAPrefilter}(\mathcal{R}_q, \mathcal{C}_q)$. This step exposes a broader range of potentially complementary notes to the selector while bounding its input.

Minimum-subset selection and grounding. An LLM receives q and $\mathcal{P}_q$ and is instructed to select the smallest memory subset that jointly supplies the required information. Its selected IDs are checked against the prefiltered pool and deduplicated; the implementation uses requirement-coverage output and a fixed fallback if the valid subset is empty (Appendix E). Only the resulting selected notes S are grounded to source turns by ϕ to produce $E_0 \subseteq \mathcal{M}$; the retriever's initial top-k is a reference cutoff, not a retained base set.

This stage improves requirement coverage within the initial candidate pool but remains bounded by that pool. A required record can therefore be absent from E_0 even when E_0 already contains several records related to the question. Stage II treats this remaining gap as an acquisition problem instead of reselecting repeatedly from the same pool. Algorithm 1 in Appendix D summarizes the selection and grounding steps.

4.3 Stage II: Complete missing support under a budget

Multi-view access to one fixed store. Text, entity, and session indices provide different access paths to the same source-grounded records in $\mathcal{M}$. At round t, each action $a = (v, c) \in A_t$ specifies a view v and a cue c. The retrievers exclude all records already contained in E_t:

$$C_t = \text{Unique}\left(\bigcup_{(v,c)\in A_t} R_v(c, \mathcal{M} \setminus E_t) \right). \tag{8}$$

The exclusion constraint makes acquisition cumulative and duplicate-free. The retention rule κ_π ranks the returned records by relevance and by novelty with respect to the current set, keeps at most k_{keep} of them, and adds them to E_t by set union. Consequently, each round either extends the reachable evidence or leaves the set unchanged.

Budgeted stopping. The budget $B = (B_{\text{step}}, B_{\text{call}}, B_{\text{new}})$ limits the number of rounds, retrieval calls, and newly retained records. The main configuration allows three rounds, six calls, and 30 new records. We use budget exhaustion as the stopping condition because the offline sufficiency measures in Section 3 require either a reference answer or gold annotations and are therefore unavailable at inference time. Algorithm 2 in Appendix D gives the shared completion loop; the fixed schedules differ only in how they instantiate A_t.

4.4 Access schedules for completion

The shared completion loop admits two fixed schedules for the two ways required support can remain outside E_0. They share FCA-MS, $\mathcal{M}$, the exclusion rule, the budget, and the answerer; only the allocation of retrieval calls changes. We also evaluate state-conditioned completion as a controlled variant of Stage II; its settings and results are reported in Section 5.2 and Appendix I.

BFR-Text: deepen the text ranking. Each round queries the text index with the original question q, removes ids already in E_t, and keeps the first $k_{\text{keep}} = 8$ unseen hits. Repeating the same query advances beyond the initial page without query rewriting. This schedule targets cases in which the missing evidence is relevant to q but appears below the first cutoff.

BFR-MV: diversify the access path. The multi-view schedule issues text, entity, and session actions in a fixed order. The text action uses q; the entity action uses the first person named in q when available; the session action uses the latest retained record as an anchor. Entity and session access can recover records that do not rank highly under the full question string.

5 Experiments

In this section, we conduct controlled experiments to evaluate BFR and answer four research questions: **(RQ1)** How does BFR compare with same-store memory baselines in answer quality? **(RQ2)** Do retrieved sets contain sufficient evidence to answer the question, and how does direct sufficiency compare with annotated-support hits? **(RQ3)** How do the two stages and access schedules contribute to performance? **(RQ4)** How do answer quality and evidence coverage change with additional retrieval cost? The appendix reports full operating-point, question-type, and cross-system results.

5.1 Experimental setup

Datasets. We evaluate on two benchmarks for long-term conversational memory. LoCoMo (Maharana et al., 2024) tests understanding of long, multi-session conversations, including questions that require linking information across turns. LongMemEval-S (Wu et al., 2025) tests whether a memory system can answer questions about information distributed across extended interaction histories and sessions. Both provide evidence annotations, allowing us to assess retrieval as well as final answers. To keep multi-arm evaluation costs manageable, we use a fixed stratified 200-question LoCoMo development subset; we evaluate all 500 questions of LongMemEval-S. Dataset provenance and the construction of every derived evaluation subset are documented in Appendix G.

Table 2: **Performance across different question types on LoCoMo** (n=200). Same flat store and extractive answerer; BFR is BFR-MV C6. F1 and LLM-Judge (J) are percentages. Bold denotes the best value in each column.

Method	Multi-Hop		Temporal		Open Domain		Single-Hop		Overall	
	F1	J	F1	J	F1	J	F1	J	F1	J
Mem0 (Chhikara et al., 2025)	29.3	42.4	74.1	78.0	2.2	0.0	58.5	59.6	55.3	59.0
A-Mem (Xu et al., 2025)	32.3	42.4	70.2	72.0	**3.8**	0.0	47.1	51.4	48.7	53.0
CoM* (Xu et al., 2026)	29.3	36.4	66.2	70.0	3.2	0.0	52.3	56.0	50.0	54.0
MRAgent-Flat* (Ji et al., 2026)	32.8	45.5	72.1	78.0	2.9	0.0	63.2	68.8	58.0	64.5
BFR (ours)	**44.6**	**48.5**	**78.1**	**80.0**	2.7	**12.5**	**75.7**	**77.1**	**68.3**	**70.5**

Table 3: **Performance across different question types on LongMemEval-S** (n=500). The four question-type columns report Judge accuracy; All values are percentages. All rows share the same turn store, extractive answerer, and judge. Bold and underline denote the best and second-best values in each column. The full six-type Turn Hit breakdown is in Appendix K.

	Judge by question type				Overall	
Method	Multi-S	Single-S	Temporal	Preference	Judge	Turn Hit
Mem0 (Chhikara et al., 2025)	<u>60.9</u>	87.1	<u>65.4</u>	26.7	<u>72.4</u>	64.8
A-Mem (Xu et al., 2025)	54.1	87.1	57.1	**33.3**	68.6	<u>82.2</u>
CoM *(Xu et al., 2026)	54.1	87.1	57.1	**33.3**	68.6	80.6
MRAgent-Flat* (Ji et al., 2026)	58.6	<u>90.0</u>	63.2	<u>30.0</u>	71.4	78.8
BFR (ours)	**69.2**	**92.9**	**84.2**	<u>30.0</u>	**82.2**	**91.4**

Baselines. We evaluate BFR against four representative memory-augmented baselines: Mem0 (Chhikara et al., 2025), A-Mem (Xu et al., 2025), CoM* (Xu et al., 2026), and MRAgent-Flat* (Ji et al., 2026). Within each benchmark, all arms share the same flat turn store, extractive answerer, and answer judge. The starred methods are controlled same-store adaptations of the published systems. Baseline retrieval and BFR initialization are detailed in Appendix E.

Metrics. Answer quality is measured by Porter stem F1 and LLM-Judge (J) on LoCoMo, and by Judge on LongMemEval-S. The answer judge is `gpt-5.5` at temperature zero with prompt `mem0_accuracy_v1`. On LoCoMo, $\mathcal{G}(q)$ is the released set of supporting-record identifiers. We report the sufficiency indicators in Section 3 and two finer measures of annotated-evidence recovery:

$$\mathrm{ER}(q,E)=\frac{|\,\mathrm{src}(E)\cap\mathcal{G}(q)|}{|\mathcal{G}(q)|},\qquad \mathrm{CER}(q,E)=\mathbb{1}[\mathcal{G}(q)\subseteq\mathrm{src}(E)]. \tag{9}$$

Evidence Recall (ER) is the fraction of annotated supporting records in $\mathcal{G}(q)$ that E recovers, while Complete Evidence Recall (CER) requires every annotated supporting record. On LongMemEval-S, $\mathcal{H}(q)$ comprises turns marked `has_answer`; Turn Hit measures whether the retrieved set reaches one. These retrieval and set-level measures complement rather than replace end-to-end answer scores. Prompts are listed in Appendix O.

5.2 Experimental Result

RQ1: Does BFR improve answer quality under the same memory store?

BFR achieves the best overall answer quality among same-store methods on both benchmarks (Table 2, Table 3). On LoCoMo, it leads in Stem F1 and Judge across Multi-Hop, Temporal, and Single-Hop questions. On LongMemEval-S, it also leads in overall Judge and Turn Hit, with a particularly strong gain on Temporal questions. The small LoCoMo Open Domain slice is reported without a separate type-level conclusion. These results are consistent with the two-stage design: FCA-MS assembles complementary initial evidence, and budgeted completion retrieves unseen support beyond the initial set. The following analyses test whether this support closes evidence

Table 4: **Set sufficiency and annotated-evidence reach across same-store systems.** On LoCoMo (n=200), LLM-Suff@Set is the blinded set-level judgment and Gold Hit measures annotated-evidence access. On LongMemEval-S, LLM-Suff@Set is the set-level judgment and Turn Hit measures annotated-evidence access over all 500 questions, with unannotated items counted as misses. All values are percentages. BFR-Text uses C3 and BFR-MV uses C6.

System	LoCoMo (n=200)		LongMemEval-S (n=500)	
	LLM-Suff@Set	Gold Hit	LLM-Suff@Set	Turn Hit
Mem0 (Chhikara et al., 2025)	45.5%	57.0%	48.0%	64.8%
A-Mem (Xu et al., 2025)	36.5%	50.0%	38.0%	82.2%
CoM* (Xu et al., 2026)	44.0%	49.0%	49.0%	80.6%
MRAgent-Flat* (Ji et al., 2026)	54.5%	62.5%	56.0%	78.8%
FCA-MS (Stage I Only)	42.5%	51.0%	48.0%	85.0%
BFR-Text	62.0%	72.5%	63.0%	88.6%
BFR-MV	**63.5%**	**73.5%**	**65.0%**	**91.4%**

Table 5: **Where budgeted completion closes the evidence gap on LoCoMo** (n=200). CER is Complete Evidence Recall; LLM-Suff@Set is the blinded set-level judgment. Text and MV denote BFR-Text at C3 and BFR-MV at C6, with mean realized calls of 3.00 and 5.99; FCA-MS uses no completion calls. Open Domain contains only eight questions and is interpreted descriptively.

Question type	n	FCA-MS		BFR-Text		BFR-MV	
		CER	LLM-Suff@Set	CER	LLM-Suff@Set	CER	LLM-Suff@Set
Multi-Hop	33	6.1	15.2	18.2	27.3	12.1	30.3
Temporal	50	66.0	60.0	82.0	74.0	82.0	74.0
Open Domain	8	12.5	0.0	12.5	0.0	25.0	0.0
Single-Hop	109	41.3	45.9	71.6	71.6	70.6	73.4
Overall	200	40.5	42.5	63.0	62.0	62.0	63.5

gaps and how stage, access schedule, and budget contribute. Paired same-store comparisons and the separate comparison with CoM and Full-MRAgent are in Appendix H and Appendix C, respectively.

RQ2: Do the retrieved sets contain the evidence needed for an answer?

BFR achieves the highest observed LLM-Suff@Set on both benchmarks among the compared same-store systems (Table 4). Under the main budget configuration, both completion schedules exceed FCA-MS alone and all baselines on set-level sufficiency and annotated-evidence coverage, with BFR-MV highest on every measure. Since FCA-MS alone trails both schedules, the gain comes from completing the initial set beyond its candidate pool rather than from selection within it.

Table 6: **Stage and access-schedule ablation.**

Configuration	LoCoMo		LongMemEval-S	
	F1	J	J	TH
FCA-MS	48.6	53.5	70.4	85.0
FCA-MS *w/ BFR-Text*	67.8	70.0	81.6	88.6
FCA-MS *w/ BFR-MV*	**68.3**	**70.5**	**82.2**	**91.4**

Table 5 further shows that completion improves complete recovery of annotated evidence on LoCoMo. However, the two measures rank the schedules differently: BFR-Text has higher overall CER, whereas BFR-MV has higher LLM-Suff@Set. On LongMemEval-S, A-Mem likewise has higher Turn Hit than Mem0 but lower set-level sufficiency. Thus, annotated-evidence access alone does not establish whether the retrieved set jointly supports the answer. Detailed LoCoMo ER, CER, and LLM-Suff@Set results are in Appendix I, and the six-type LongMemEval-S Turn Hit breakdown is in Appendix K.

RQ3: Which stage and access schedule produce the gains?

FCA-MS improves annotated-evidence reach over one-shot ranking on both benchmarks; budgeted completion from this initializer yields further gains in evidence access and answer quality. In Figure 4, FCA-MS raises LoCoMo Gold Hit from 50.0% to 51.0% and LongMemEval-S Turn Hit from 82.2% to 85.0%. Adding either BFR schedule retrieves more annotated support at the reported operating points. Table 6 shows that, relative to FCA-MS alone, both schedules also raise Stem F1 and Judge on LoCoMo and Judge on LongMemEval-S. The schedules favor different reported operating points. At the matched C3 point in figure 4, BFR-Text has higher LoCoMo Gold Hit than BFR-MV. The full BFR-MV C6 configuration has the highest LongMemEval-S Judge and Turn Hit in Table 6, at a higher call budget than BFR-Text C3. Type-level results are in Table 13 and Appendix K.

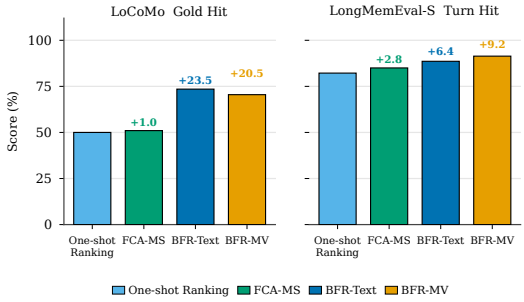


Figure 4: **Coverage relative to one-shot ranking.**

RQ4: What does additional retrieval cost buy?

Figure 5 shows the cost–quality tradeoff. The panels show BFR-Text and BFR-MV starting from the same FCA-MS C0 set. The horizontal axis is the mean realized number of retrieval calls; ER and CER measure partial and complete recovery of annotated evidence, respectively. Lines connect observed operating points only. BFR-Text C4 and C6 coincide at four realized calls. Most of the answer-quality gain over FCA-MS appears after the first completion call. At the matched three-call point, BFR-Text exceeds BFR-MV on all four plotted metrics. BFR-MV reaches its highest scores at C9, but requires nearly nine calls, and its curves are not monotonic between budgets. The C6 configuration in the main tables remains the fixed default operating point rather than a sweep-selected maximum. The complete numerical grid, including evidence-set size and LLM-Suff@Set, is in Appendix I; the matched-call RSC comparison is in Appendix L.

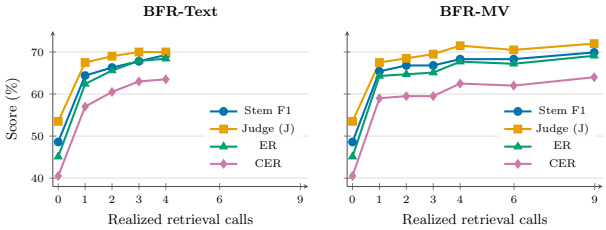


Figure 5: **Answer quality and annotated-evidence recovery versus retrieval cost on LoCoMo** (n=200).

6 Conclusion and Discussion

In this paper, we proposed BFR, a two-stage method for building sufficient evidence sets that can support answers in long-term memory QA. On LoCoMo and LongMemEval-S, budgeted completion alone raises annotated-evidence reach over one-shot ranking, and the full two-stage framework further improves evidence coverage and judged answer quality over same-store baselines. Ablations attribute most of the gain to broader evidence access rather than adaptive state control, and human checks show that gold-free sufficiency estimates remain too unreliable to serve as stopping criteria. More broadly, these results motivate a different objective for long-term memory retrieval: not to rank individually relevant memories, but to construct a set whose members collectively satisfy the information requirements of the question. From this view, retrieval involves both *composition*, assembling complementary support within reach, and *completion*, continuing beyond the initial retrieval boundary when necessary. This makes set-level sufficiency a natural bridge between memory access and downstream reasoning: retrieval should deliver an answer-supporting evidentiary record, not simply a list of similar items. As agent histories become longer and required evidence is increasingly distributed across sessions, we expect this distinction between relevance and sufficiency to become central to how long-term memory systems are designed and evaluated.

AI USE STATEMENT

Generative AI systems were used in two roles. As part of the reported method and evaluation, language models perform requirement decomposition, candidate–requirement coverage judgments, answer synthesis, answer judging, and blinded LLM-Suff@Set labeling and adjudication; the corresponding roles, prompts, and evaluation rules are documented in the paper and appendix. Cursor and OpenAI Codex also assisted with experimental planning, code generation and debugging, result organization, literature discovery, and manuscript drafting and language editing. The authors inspected AI-produced code and outputs, reran the reported computations, verified citations and numerical claims against stored experiment artifacts, and made all final methodological and editorial decisions. Public benchmark conversations, questions, and reference answers were not generated or altered by these tools. The authors take responsibility for the complete contents of the submission.

REFERENCES

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

Yuyang Hu, Shichun Liu, Yanwei Yue, Guibin Zhang, Boyang Liu, Fangyi Zhu, Jiahang Lin, Honglin Guo, Shihan Dou, Zhiheng Xi, et al. Memory in the age of ai agents. *arXiv preprint arXiv:2512.13564*, 2025.

Shuo Ji, Yibo Li, and Bryan Hooi. Memory is reconstructed, not retrieved: Graph memory for LLM agents. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026.

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

Jiazheng Kang, Mingming Ji, Zhe Zhao, and Ting Bai. Memory OS of AI agent. *arXiv preprint arXiv:2506.06326*, 2025.

Richard Lempert. The new evidence scholarship: Analyzing the process of proof. *BUL Rev.*, 66: 439, 1986.

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

Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. 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)*, pp. 13851–13870, Bangkok, Thailand, 2024. Association for Computational Linguistics.

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

Zhuoshi Pan, Qianhui Wu, Huiqiang Jiang, Xufang Luo, Hao Cheng, Dongsheng Li, Yuqing Yang, Chin-Yew Lin, H. Vicky Zhao, Lili Qiu, and Jianfeng Gao. SeCom: On memory construction and retrieval for personalized conversational agents. In *The Thirteenth International Conference on Learning Representations*, 2025.

Michael S. Pardo. What makes evidence sufficient? *Arizona Law Review*, 65(2):431–478, 2023.

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

Zhen Tan, Jun Yan, I-Hung Hsu, Rujun Han, Zifeng Wang, Long Le, Yiwen Song, Yanfei Chen, Hamid Palangi, George Lee, et al. In prospect and retrospect: Reflective memory management for long-term personalized dialogue agents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8416–8439, 2025.

Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. Longmemeval: Benchmarking chat assistants on long-term interactive memory. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=pZiyCaVuti.

Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. A-MEM: Agentic memory for LLM agents. In *Advances in Neural Information Processing Systems*, 2025.

Xiucheng Xu, Bingbing Xu, Xueyun Tian, Zihe Huang, Rongxin Chen, Yunfan Li, and Huawei Shen. Chain-of-memory: Lightweight memory construction with dynamic evolution for LLM agents. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics*, 2026.

Zeyu Zhang, Quanyu Dai, Xiaohe Bo, Chen Ma, Rui Li, Xu Chen, Jieming Zhu, Zhenhua Dong, and Ji-Rong Wen. A survey on the memory mechanism of large language model-based agents. *ACM Transactions on Information Systems*, 43(6):1–47, 2025.

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

CONTENTS

1 Introduction **1**

2 Related Work **3**

3 WHAT COUNTS AS A SUFFICIENT CANDIDATE MEMORY SET **3**

4 BFR: Composing and Completing Sufficient Evidence **4**

4.1 Problem: construct a sufficient set under a retrieval budget 4

4.2 Stage I: Compose complementary evidence with FCA-MS 5

4.3 Stage II: Complete missing support under a budget 6

4.4 Access schedules for completion 6

5 Experiments **6**

5.1 Experimental setup 6

5.2 Experimental Result 7

6 Conclusion and Discussion **9**

A Evidence-law mapping **14**

B Case trace for figure 1 **14**

C Structured memory boundary **14**

D Algorithmic specification **15**

E Implementation details **16**

F Data usage and sealing **17**

G Dataset provenance and derived evaluation sets **17**

H Same-store LoCoMo controls **18**

I Budget sweep **19**

J Development robustness **21**

K Full LongMemEval-S type breakdown **21**

L Comparison with learned residual set completion **21**

M Annotation protocol **22**

N Metric definitions **22**

O Prompts **22**

P Matched three-call comparison of BFR-MV and BFR-Text **24**

A EVIDENCE-LAW MAPPING

Pardo asks what makes evidence sufficient, and answers by separating two judgments that legal fact-finding already treats as distinct (Pardo, 2023). *Relevance* is a property of an item: an exhibit is relevant when it has a bearing on a fact of consequence. *Sufficiency* is a property of a record: the body of evidence is sufficient when a reasonable finder of fact could make the finding from that record as assembled. The finder does not score items independently and then threshold the sum. The question is whether the assembled exhibits, taken together, could support one explanation of the disputed fact over its competitors. Pardo therefore emphasizes *explanatory facts*: relations between exhibits and the elements of a finding, rather than a per-item probability that an exhibit is "about" the claim.

Memory QA uses the same two objects. A memory $m \in \mathcal{M}$ is an exhibit, and $R(q, m)$ denotes its item-wise relevance to question q: does this memory have any tendency to matter for q? The retrieved set E is the evidentiary record. The ideal property $S(q, E)$ in equation 1 adapts Pardo's reasonable-finder test to conversational memory; the blinded LLM label $\widehat{S}_{\mathrm{LLM}}(q, E)$ in equation 2 is its reference-conditioned operational measurement. The shutter-release case in figure 1 is exactly this split: the February 5 order is a relevant exhibit, and the photography memories around it are also relevant, but the record cannot support the finding "five days" until the February 10 arrival enters E.

The method in Section 4 implements the two operations that this split requires. Information requirements $\mathcal{R}_q = \{r_j\}$ are the explanatory elements of the finding: order date and arrival date in the running case; a fact and a bridge in multi-hop questions; the items of a list. The requirement-conditioned candidate assessment in Stage I concerns relations between an exhibit and a needed element, rather than a single query–memory score. Stage I composition is the assembly of a record whose exhibits jointly cover those elements. Stage II completion is what a finder does when the current record is insufficient: further exhibits are acquired, from the same store, rather than re-ranking exhibits already known to be relevant. The exclusion $K_t \subseteq \mathcal{M} \setminus E_t$ in equation 6 is the legal analogue of not re-admitting the same exhibit. Gold Hit G and Turn Hit ask whether an annotated answer-bearing exhibit has entered the record; they correspond to coverage of at least one marked element, which is why Section 3 treats them as proxies associated with S rather than as the sufficiency test itself.

This mapping is conceptual, not a claim that agent memory should mimic courtroom procedure. It specifies the object we optimize: a record that could support a reasonable finding, constructed by composing complementary exhibits and completing the record when it cannot.

B CASE TRACE FOR FIGURE 1

Figure 1 uses question `b3c15d39` from LONGMEMEVAL-S. The answer requires two user turns: the remote shutter release was ordered on February 5 and arrived on February 10. Both lexical top-k and FCA-MS reach the order turn but miss the arrival turn, leaving evidence recall at 50% and producing an incorrect answer. BFR-MV reaches both turns and answers "5 days" under the same extractive answerer and judge.

Table 7: **Trace behind the relevance–sufficiency case.** Case evidence recall counts the two required turns; the first two sets contain relevant photography memories but miss one required date.

Access	Order date	Arrival date	Gold recall	Judge
Lexical top-k	✓	—	50%	Wrong
FCA-MS (Stage I)	✓	—	50%	Wrong
BFR-MV	✓	✓	100%	Correct

C STRUCTURED MEMORY BOUNDARY

This appendix reports the cross-system comparison that was removed from the main LOCOMO table. The slice is the conv-30 overlap (n=81) under a unified extractive answerer and judge. BFR-

Table 8: **Cross-system boundary on the LoCoMo conv-30 overlap** (n=81), unified extractive answerer and judge; † marks rows re-scored under this protocol. Bold / underline: best / second-best. Open Domain is absent from the slice.

	Multi-Hop		Temporal		Single-Hop		Overall	
Method	F1	J	F1	J	F1	J	F1	J
BFR-Text†	**41.4**	54.5	**100.0**	**100.0**	<u>75.5</u>	77.3	**78.7**	81.5
CoM† (Xu et al., 2026)	<u>41.2</u>	<u>63.6</u>	<u>96.2</u>	<u>96.2</u>	71.8	<u>84.1</u>	75.5	<u>85.2</u>
MRAgent-Flat† (Ji et al., 2026)	12.9	45.5	92.3	92.3	62.3	68.2	65.2	72.8
Full-MRAgent† (Ji et al., 2026)	26.5	**90.9**	92.3	<u>96.2</u>	**78.8**	**95.5**	<u>76.1</u>	**95.1**

Table 9: **Flat acquisition vs. structured memory** on the 81-question overlap slice under a shared extractive answerer and judge. Full MRAgent and CoM change the store and are reported as a boundary, not as fair same-substrate baselines.

Configuration	Store	Gold Hit (%)	Judge Acc (%)
MRAgent-Flat	flat turns	67.9	72.8
BFR-Text	flat turns	77.8	81.5
CoM	organized pool	80.2	85.2
Full MRAgent	Cue–Tag–Content graph	**88.9**	**95.1**

Text and MRAgent-Flat still read flat turns; CoM uses an organized candidate pool; Full-MRAgent uses a Cue–Tag–Content graph. Open Domain is absent from this slice. The comparison is a boundary, not a fair same-store baseline: the store, derived facts, and tools move together.

Table 9 reports the same slice by store type. Full-MRAgent reaches 88.9% Gold Hit and 95.1% judged accuracy, while MRAgent-Flat reaches 67.9% / 72.8% and BFR-Text reaches 77.8% / 81.5%. CoM reaches 80.2% / 85.2% under the same unified answerer.

Paired against BFR-Text, Full-MRAgent reaches gold on 12 questions that BFR-Text misses; the reverse occurs on 3 questions. Comparing Full-MRAgent with MRAgent-Flat, the full pipeline is judged correct alone on 20 questions, while the flat variant is correct alone on 2. Mean evidence-set sizes are 2.4 (Full), 14.2 (Flat), and 34.0 (BFR-Text), so the graph pipeline reaches gold more often while retaining far less text. This pattern is consistent with associative precision rather than search volume alone, but the store, tools, and derived facts still move together. A causal graph-edge claim would require toggling the graph under identical tools.

D Algorithmic specification

Algorithm 1 specifies how Stage I constructs the initial evidence set, and algorithm 2 gives the Stage II completion loop shared by the fixed access schedules.

Algorithm 1 Stage I: Formal Concept Analysis for Memory Selection (FCA-MS)

Require: Question q, agent-memory notes $\widetilde{\mathcal{M}}$, source store $\mathcal{M}$, usual retrieval cutoff k, expanded-pool size $N > k$, prefilter cap P, grounding limit b, fallback size f
Ensure: Source-grounded initial evidence set E_0

$\mathcal{C}_q \leftarrow \mathrm{TopN}(\mathrm{Retrieve}_{\mathrm{agent}}(q, \widetilde{\mathcal{M}}), N)$ ▷ Search beyond the usual top-k
$\mathcal{R}_q \leftarrow \mathrm{Decompose}(q)$ ▷ Identify facts, bridges, list items, and temporal constraints
$\mathcal{P}_q \leftarrow \mathrm{FCAPrefilter}(\mathcal{R}_q, \mathcal{C}_q, P)$ ▷ Reduce the expanded pool to at most P notes
$(\widehat{S}, \Gamma) \leftarrow \mathrm{LLMMinSubset}(q, \mathcal{P}_q)$ ▷ Request selected IDs and requirement coverage
$S \leftarrow \mathrm{Sanitize}(\widehat{S}, \Gamma, \mathcal{P}_q)$ ▷ Keep valid IDs once; use coverage order if needed
if $S = \varnothing$ **then**
    $S \leftarrow \mathrm{First}_f(\mathcal{P}_q)$ ▷ Fall back to the prefilter head
end if
$E_0 \leftarrow \mathrm{PositiveOverlapTop}_b(S, \mathcal{M})$ ▷ Score turns; keep positive scores in stable order
return E_0 ▷ Pass the composed evidence set to Stage II

Algorithm 2 Stage II: Budgeted Flat Reconstruction (BFR) Completion

Require: Question q, initial evidence E_0, flat store $\mathcal{M}$, schedule $\pi \in \{\text{BFR-Text}, \text{BFR-MV}\}$, budget $B = (B_{\mathrm{step}}, B_{\mathrm{call}}, B_{\mathrm{new}})$, keep limit k
Ensure: Completed evidence set E_T

$E \leftarrow E_0$; $t \leftarrow 0$; $b_{\mathrm{call}} \leftarrow 0$; $b_{\mathrm{new}} \leftarrow 0$ ▷ Initialize the cumulative evidence state and counters
while $t < B_{\mathrm{step}}$ and $b_{\mathrm{call}} < B_{\mathrm{call}}$ and $b_{\mathrm{new}} < B_{\mathrm{new}}$ **do**
    if $\pi =$ BFR-Text **then**
        $A_t \leftarrow \{(\mathrm{text}, q)\}$ ▷ Advance beyond the current text-ranking cutoff
    else
        $A_t \leftarrow \{(\mathrm{text}, q), (\mathrm{entity}, \mathrm{FirstPerson}(q)), (\mathrm{session}, \mathrm{Latest}(E))\}$ ▷ Diversify access through text, entity, and session cues
    end if
    $C_t \leftarrow \varnothing$ ▷ Hold the current evidence fixed throughout this round
    for all $(v, c) \in A_t$ in schedule order **do**
        if $c = \varnothing$ or $b_{\mathrm{call}} = B_{\mathrm{call}}$ **then**
            continue ▷ Skip unavailable cues or exhausted call budgets
        end if
        $C_t \leftarrow C_t \cup R_v(c, \mathcal{M} \setminus E)$
        $b_{\mathrm{call}} \leftarrow b_{\mathrm{call}} + 1$ ▷ Collect candidates relative to the same round-start evidence
    end for
    $C_t \leftarrow \mathrm{Unique}(C_t)$
    $K_t \leftarrow \kappa_\pi(C_t, q, E, k)$ ▷ Apply the retention rule once to the merged candidates
    $K_t \leftarrow \mathrm{ClipToBudget}(K_t, B_{\mathrm{new}} - b_{\mathrm{new}})$
    $E \leftarrow E \cup K_t$; $b_{\mathrm{new}} \leftarrow b_{\mathrm{new}} + |K_t|$ ▷ Update evidence after the round's candidate union and retention
    $t \leftarrow t + 1$ ▷ Proceed until an observable budget is exhausted
end while
return $E_T \leftarrow E$ ▷ Supply the completed set to the fixed answerer

E IMPLEMENTATION DETAILS

Same-store baselines. Within each benchmark, the flat conversational turn pool, extractive answerer, and Mem0-style judge are fixed across arms. Mem0 uses BM25-lite over the turn pool; A-Mem uses dense top-k over the same pool without note evolution. CoM* uses MiniLM candidates with anchor, chain, and blocking. MRAgent-Flat* restricts its tools to flat turns, omitting the Cue–Tag–Content graph and derived facts. The starred methods are controlled same-store adaptations, not the official full systems.

FCA-MS initialization. In the LoCoMo pilot, dense A-Mem note retrieval yields $N = 50$ candidates. `fca_prefilter_candidates` uses the decomposed requirements to retain at most $P = 20$ notes. The `setr_pool` v1 prompt asks an LLM for the smallest subset that jointly answers the question, with no fixed note-count cap. Invalid or duplicate IDs are removed; an empty selection uses requirement-coverage output and then the prefilter head as fallback. The resulting sets contain 1–5 notes (mean 1.86). Only these notes enter `map_amem_ids_to_turns(..., top_n=10)`, which ranks source turns by positive token overlap with the selected note content and context. All 200 pilot questions yielded 10 mapped turns; these turn IDs form E_0 for Stage II. A deployment with direct source pointers can replace this approximate alignment with an exact grounding map ϕ.

Budgeted completion. BFR-Text is recorded as `IterRAG-Text` in experiment logs. It resubmits the original question to the text index, excludes current evidence ids, and retains the first eight unseen results. BFR-MV is recorded as `Q-Fixed`; it runs the fixed text→entity→session schedule in Section 4.4. The default B_{mid} point allows 3 rounds, 6 retrieval calls, and 30 newly retained records. This repeated, exclusion-aware schedule generalizes one-pass acquisition by continuing to collect unseen support until the observable budget is exhausted.

F Data usage and sealing

All 10 LoCoMo conversations were visible during development, so every LoCoMo number measures stability across conversations rather than generalization to unseen data. LongMemEval-S was not read, scored, or inspected until the LoCoMo protocol, arm definitions, budget grid, and metrics were frozen. After the initial BFR evaluation on LongMemEval-S, CoM* and MRAgent-Flat* were evaluated under the same flat-store protocol and included in the main comparison.

G Dataset provenance and derived evaluation sets

Public source benchmarks. LoCoMo was introduced by Maharana et al. (2024) as a benchmark for very long-term conversational memory. Its released QA corpus contains ten long, multi-session conversations produced by the authors' persona- and event-conditioned conversation-generation framework. The release pairs questions and reference answers with category labels and, when available, the dialogue-turn identifiers that support the answer. We use the released conversations and QA annotations; we do not generate new dialogue content or alter the reference answers.

LongMemEval was introduced by Wu et al. (2025) using an attribute-controlled pipeline that places evidence sessions inside coherent, timestamped conversation histories. We use the official cleaned LongMemEval-S file, which contains 500 questions covering single-session user and assistant facts, preferences, multi-session reasoning, temporal reasoning, and knowledge updates. The complete 500-question split is evaluated. Turn Hit and turn recall use the official answer-bearing-turn fields; 479 questions contain at least one such turn. The main Turn Hit results use all 500 questions and count the 21 questions without an annotated answer turn as misses. The matched-C3 analysis in Appendix P instead reports Turn Hit over the 479 questions with nonempty answer-turn annotations.

Derived LoCoMo evaluation sets. The 200-question pilot is a fixed, stratified subset drawn from the official ten conversations and shared by every same-store arm. The pilot was sampled with seed 20260908 and contains 33 Multi-Hop, 50 Temporal, 8 Open Domain, and 109 Single-Hop questions, using the released manifest labels without reclassification. It is a development subset for controlled method, ablation, annotation, and budget comparisons, not an official held-out split. The 1,536-question gold subset contains all questions in our LoCoMo release with usable supporting dialogue-turn identifiers; it is used only for development robustness across conversations. The conv-30 slice contains the 81 question ids with aligned outputs for the cross-system comparison in Appendix C. It is an overlap set rather than an independently sampled benchmark, and Open Domain is absent from this slice.

Evidence-state and budget artifacts. For method-aligned sufficiency analysis, seven arms (Mem0, A-Mem, CoM*, MRAgent-Flat*, FCA-MS, BFR-Text C3, and BFR-MV C6) are evaluated

on each of the fixed 200 questions by the same blinded LLM-Suff@Set protocol in Appendix M. These 1,400 method–question evaluations concern retrieval outputs rather than additional conversations. The budget grid applies the fixed BFR access policies and call caps to the same pilot; identical evidence states are deduplicated by question and evidence hash. These artifacts evaluate retrieval behavior over the public benchmark; they are not presented as new datasets.

Table 10: **Provenance of the public benchmarks and derived evaluation sets.** Only LoCoMo and LongMemEval-S are independently released benchmarks; all other rows are fixed subsets or evidence states constructed from them.

Evaluation set	Parent	Size	Construction and role
LoCoMo release	Public benchmark	10 conversations	Official multi-session conversations, QA pairs, categories, and supporting turn ids.
LoCoMo pilot	LoCoMo	200 questions	Fixed stratified development subset shared by all same-store arms.
LoCoMo gold subset	LoCoMo	1,536 questions	All questions with usable gold turn ids; used for development robustness.
conv-30 overlap	LoCoMo	81 questions	Qid intersection for cross-system scoring; not a held-out split.
LLM-Suff@Set evaluations	LoCoMo pilot	1,400 arm–question pairs	200 questions paired with Mem0, A-Mem, CoM*, MRAgent-Flat*, FCA-MS, BFR-Text C3, and BFR-MV C6 under blinded method identities.
Budget grid	LoCoMo pilot	Fixed policy–budget runs	BFR-Text and BFR-MV call budgets evaluated before evidence-hash deduplication.
LongMemEval-S	Public benchmark	500 questions	Official cleaned small split, evaluated in full after protocol freezing.

H SAME-STORE LOCOMO CONTROLS

Table 11: **Retrieval cost of the adapted same-store controls** on the fixed 200-question pilot. Calls and evidence-set size are realized means.

Method	Gold Hit	Calls	Evidence-set size
CoM*	49.0	1.00	10.0
MRAgent-Flat*	62.5	2.54	11.8

Table 12: **Paired LoCoMo comparisons against the adapted controls.** Δ is the first method minus the second. Judge uses two-sided exact McNemar; Stem F1 uses a 10,000-sample question-level paired bootstrap (seed 20260908).

Comparison	ΔJ	p	Δ F1	95% CI
BFR-Text − CoM*	+16.0	1.4×10^{-5}	+17.8	[11.3, 24.2]
BFR-MV − CoM*	+16.5	8.7×10^{-6}	+18.2	[11.8, 24.8]
BFR-Text − MRAgent-Flat*	+5.5	0.071	+9.8	[5.0, 14.8]
BFR-MV − MRAgent-Flat*	+6.0	0.043	+10.2	[5.5, 15.2]

CoM* completes all 200 questions. MRAgent-Flat* completes 199; `q_00208` has empty evidence and remains in the denominator as an empty prediction. These rows are distinct from the 81-question full-system comparison in Appendix C.

I BUDGET SWEEP

Table 13: **BFR access schedules by question type on LoCoMo** (n=200). F1 and J are percentages; Text and MV use the C3 and C6 operating points, respectively.

	Multi-Hop		Temporal		Open Domain		Single-Hop		Overall	
Schedule	F1	J	F1	J	F1	J	F1	J	F1	J
BFR-Text	44.7	54.5	78.1	80.0	1.9	0.0	74.9	75.2	67.8	70.0
BFR-MV	44.6	48.5	78.1	80.0	2.7	12.5	75.7	77.1	68.3	70.5

Table 14: **Evidence Recall and Complete Evidence Recall by question type on the LoCoMo pilot.** Text and MV use the C3 and C6 operating points, respectively.

		FCA-MS		BFR-Text		BFR-MV	
Question type	n	ER	CER	ER	CER	ER	CER
Multi-Hop	33	24.8	6.1	**42.6**	**18.2**	41.8	12.1
Temporal	50	68.0	66.0	**82.0**	**82.0**	**82.0**	**82.0**
Open Domain	8	18.8	12.5	27.1	12.5	**31.2**	**25.0**
Single-Hop	109	42.7	41.3	**72.0**	**71.6**	70.6	70.6
Overall	200	45.1	40.5	**67.9**	**63.0**	67.2	62.0

Table 15: **Blinded LLM-Suff@Set by question type on the LoCoMo pilot.** Differences use BFR minus FCA-MS. Open Domain contains only eight questions.

Type	n	FCA	Text	MV	ΔText	ΔMV
Multi-Hop	33	15.2	27.3	**30.3**	+12.1	+15.2
Temporal	50	60.0	**74.0**	**74.0**	+14.0	+14.0
Open Domain	8	0.0	0.0	0.0	0.0	0.0
Single-Hop	109	45.9	71.6	**73.4**	+25.7	+27.5
Overall	200	42.5	62.0	**63.5**	+19.5	+21.0

Overall, BFR-Text − FCA-MS is $+4.0$ points ($p = 0.332$), while BFR-MV − FCA-MS is 0.0 points ($p = 1.000$). No type-specific LLM-Suff@Set difference is significant after Holm correction.

Table 16: **Matched-budget quality on the LoCoMo pilot** (n=200). Calls and set size are realized means; size counts grounded source turns. FCA-MS selects 1–5 notes (mean 1.86) before mapping to 10 turns. J: judged accuracy; CER: Complete Evidence Recall.

Method	Budget	Calls	Size	F1	J	LLM-Suff@Set	CER
FCA-MS	C0	0.00	10.00	48.6	53.5	42.5	40.5
BFR-MV	C1	1.00	21.91	65.4	67.5	57.5	59.0
	C2	2.00	29.05	66.8	68.5	58.0	59.5
	C3	3.00	29.84	66.8	69.5	58.0	59.5
	C4	4.00	39.72	68.3	71.5	61.0	62.5
	C6	5.99	39.88	68.3	70.5	63.5	62.0
	C9	8.98	59.23	69.9	72.0	62.5	64.0
BFR-Text	C1	1.00	18.00	64.4	67.5	56.0	57.0
	C2	2.00	26.00	66.3	69.0	59.5	60.5
	C3	3.00	34.00	67.8	70.0	62.0	63.0
	C4/C6	4.00	40.00	69.3	70.0	62.5	63.5

Table 16 gives the full operating-point grid discussed in Section 5.2. Calls are realized means rather than caps. Across the full grid, BFR-Text occupies the three-call frontier for Judge, F1, LLM-Suff@Set, and Complete Evidence Recall, while BFR-MV reaches the highest Judge score at C9 and contributes non-dominated Judge points at C4 and C9. These operating points make the acquisition-cost frontier explicit. The source-level Pareto classification uses realized calls and evidence-set size and treats higher quality and lower cost as preferred. Table 17 contrasts the Stage-I evidence set with the completed text and multi-view schedules on both benchmarks. Table 18 reports the state-conditioned controllers from Section 4.4; their results motivate the simpler fixed completion schedules used in the main framework.

Table 17: **Annotated-evidence reach before and after completion.** LoCoMo reports Gold Hit and LongMemEval-S reports Turn Hit. The one-shot LongMemEval-S cell lists FCA-MS / Mem0; parenthesized gains use FCA-MS.

Step	LoCoMo Gold Hit	LME-S Turn Hit
One-shot (FCA-MS / Mem0)	51.0	85.0 / 64.8
BFR-MV	73.5 (+22.5)	91.4 (+6.4)
BFR-Text (vs. one-shot)	72.5 (+21.5)	88.6 (+3.6)

Table 18: **Evidence-state ablations** on the LoCoMo pilot at B_{mid}. Brackets: clustered 95% CI. Budgeted access drives the gain; state control does not. CI for Text vs. Initial $[19.1, 26.7]$; State vs. MV $[-0.8, 4.5]$.

Factor	Variant	G (%)	Δ
Access	Initial	51.0	ref.
	BFR-Text	**72.5**	$+21.5$
State	BFR-MV	73.5	ref.
	BFR-State	73.0	-0.5
	BFR-Adaptive	73.0	-0.5

BFR-State adds a missing-requirement cue and coverage/bridge retention; BFR-Adaptive also selects views and stops early from the evidence state. At B_{mid} on LoCoMo, both reach 73.0%

Gold Hit versus 73.5% for fixed BFR-MV. The clustered 95% CI for BFR-State minus BFR-MV is $[-0.8, 4.5]$ points, so this ablation does not establish a gain over fixed access.

J DEVELOPMENT ROBUSTNESS

On the broader development-contact LOCOMO set (n=1536), conversation-level five-fold analysis raises Gold Hit from 54.8% for initial retrieval to 78.0% and 78.1% for text and multi-view completion.

Table 19: **Development robustness on all LOCOMO questions with gold ids** (n=1536). Conversation-level 5-fold; all conversations had development contact.

Method	Gold Hit (%) ↑
Initial retrieval	54.8
BFR-Text	78.0
BFR-MV	78.1

K FULL LONGMEMEVAL-S TYPE BREAKDOWN

The six released types show the multi-session and temporal retrieval-coverage pattern behind the answer-quality results in Table 3.

Table 20: **Full type breakdown under the unified flat-store protocol on LONGMEMEVAL-S** (all 500 questions; unannotated answer-turn items count as misses). Upd.: knowledge-update; Mul.: multi-session; Tmp.: temporal; Pref.: preference; Asst./User: single-session. The six type columns and G report Turn Hit; J is judged accuracy over all questions.

Method	Upd.	Mul.	Tmp.	Pref.	Asst.	User	G	J
Mem0	78.2	53.4	61.7	16.7	87.5	80.0	64.8	72.4
A-Mem	88.5	82.0	75.2	70.0	98.2	81.4	82.2	68.6
CoM	88.5	81.2	72.9	63.3	98.2	78.6	80.6	68.6
MRAgent-Flat	79.5	75.9	78.9	60.0	91.1	81.4	78.8	71.4
FCA-MS	89.7	84.2	80.5	70.0	100.0	84.3	85.0	70.4
BFR-Text	91.0	85.7	88.7	70.0	100.0	90.0	88.6	81.6
BFR-MV	92.3	91.0	91.0	76.7	100.0	91.4	91.4	82.2

L COMPARISON WITH LEARNED RESIDUAL SET COMPLETION

Table 21: **Learned residual set completion (RSC) comparison** on the LOCOMO pilot (n=200). The three completion policies use a matched budget of three retrieval calls; initial retrieval is shown as a zero-call reference.

Method	Gold Hit (%) ↑	Calls
Initial retrieval	51.0	0.00
BFR-Text C3	**72.5**	3.00
BFR-MV C3	<u>70.5</u>	3.00
RSC (dense residual)	58.0	3.00

Under the same three-call budget, the fixed text and multi-view schedules recover more annotated gold evidence than RSC: Gold Hit is 72.5% for BFR-Text C3, 70.5% for BFR-MV C3, and 58.0%

for RSC. Under the current training signal and flat-store protocol, learning a residual query therefore does not improve over the fixed completion schedules.

M ANNOTATION PROTOCOL

Suff@Set is labelled on a question–evidence snapshot rather than on the system answer. The blinded input contains the question, reference answer, and anonymized evidence text; it withholds the method, budget, system answer, and gold evidence identifiers. Two independent judges (`gpt-5.5` and `claude-sonnet-4-6`) assign *sufficient*, *insufficient*, or *uncertain*; `gpt-6-sol` adjudicates disagreements. All calls use temperature zero. Because the protocol includes the reference answer and uses model judges, Suff@Set is an LLM-only, reference-conditioned offline evaluation measure rather than a human annotation or an inference-time stopping signal. An uncertain or missing final label is excluded from the binary-rate denominator; the originally reported FCA-MS, BFR-Text C3, and BFR-MV C6 pilot produced no uncertain labels. No human Suff@Set labels are reported.

N METRIC DEFINITIONS

The sufficiency-oriented measurements used in the main text are defined here because they answer different questions about the retrieved set.

Set sufficiency S. $S(q, E) = 1$ when a careful reader can determine $y^\star(q)$ from the question and the retrieved evidence text alone (equation 1). The blinded LLM protocol estimates this property through $\widehat{S}_{\mathrm{LLM}}(q, E)$ (equation 2; Appendix M). S is a reference-conditioned record-level test; it is not computed from similarity ranks or gold evidence identifiers.

Gold Hit G. Gold Hit is 1 when the source identifiers in E intersect the annotated supporting-record identifiers (equation 3). G answers a narrower question than S: did at least one marked support record enter the context? A hit can still leave a compositional question underspecified, so we do not treat G as sufficiency.

Evidence Recall and Complete Evidence Recall. For questions with a nonempty annotated gold set, Evidence Recall is the fraction of gold identifiers in $\mathrm{src}(E)$; Complete Evidence Recall requires all gold identifiers to be present (equation 9). The former measures partial recovery; the latter requires all annotated support records. Complete Evidence Recall concerns completeness relative to benchmark annotations and is not identical to semantic Suff@Set.

Turn Hit. Turn Hit is 1 when the source identifiers in E include any annotated answer-bearing turn (equation 4). Main LONGMEMEVAL-S results average over all 500 questions, counting unannotated items as misses; Appendix P separately uses the 479 questions with answer-turn annotations. Like Gold Hit, it measures access to marked evidence rather than support of the complete answer.

Answer scores. Stem F1 follows the LoCoMo / MRAgent normalization with Porter stemming. Judge accuracy is the Mem0-style CORRECT/WRONG decision under `gpt-5.5` at temperature 0 (Appendix O). Native free-form scores from other systems are never mixed into same-store rankings without the unified-answerer note.

O PROMPTS

We list the prompts used in our evaluation pipeline. Formatting follows the prompt-box style common in recent memory-agent papers.

LLM-Judge Prompt (Mem0-style)

Your task is to label an answer to a question as 'CORRECT' or 'WRONG'. You will be given the following data:

- a question (posed by one user to another user),
- a 'gold' (ground truth) answer,
- a generated answer

which you will score as CORRECT/WRONG.

The point of the question is to ask about something one user should know about the other user based on their prior conversations. The gold answer will usually be a concise and short answer that includes the referenced topic. 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 will be a specific date, month, year, etc. The generated answer might be much longer or use relative time references, but you should be generous with your grading—as long as it refers to the same date or time period as the gold answer, it should be counted as CORRECT. Even if the format differs (e.g., "May 7th" vs "7 May"), consider it CORRECT if it is the same date.

Now it's time for the real question:

```
Question: {question}
Gold answer: {gold_answer}
Generated answer: {generated_answer}
```

First, provide a short (one sentence) explanation of your reasoning, then finish with CORRECT or WRONG. Do NOT include both CORRECT and WRONG in your response.

Just return the label CORRECT or WRONG in a json format with the key as "label":

```
{"label": "CORRECT"}
```

Suff@Set Judging Inputs and Decision Rule

You will see a question, its reference answer, and an anonymized retained evidence set from a conversational memory store. The method, budget, system answer, and gold memory identifiers are **not** shown.

Task: Decide whether the evidence text supports the complete reference answer.

- Label `sufficient` if the complete answer is supported by the provided evidence.
- Label `insufficient` if a required fact, bridge, or temporal detail is missing.
- Label `uncertain` only when the evidence does not permit a reliable decision.
- Do not use outside knowledge beyond what is written in the evidence.

Two judges apply this rule independently; a third model adjudicates disagreements.

Extractive Answerer Cue

Given the question and the ordered evidence snippets, produce a short answer string that is supported by the evidence. Prefer spans that appear in the evidence. If multiple candidates exist, choose the most specific span that answers the question. Do not invent entities that are absent from the evidence set.

BFR Multi-View Schedule (deterministic)

At each round t, issue up to three retrieval actions in fixed order:

1. **text**: cue = the original question q.
2. **entity**: cue = first person name extracted from q (skip if none).
3. **session**: cue = most recently retained evidence record (skip if none).

Exclude ids already in E_t. Retain policy-selected unseen candidates and union into E_{t+1}. Stop when the call / round / retention budget is exhausted.

P MATCHED THREE-CALL COMPARISON OF BFR-MV AND BFR-TEXT

Both policies start from the same FCA-MS initialization and make 3.00 retrieval calls. Differences are BFR-MV minus BFR-Text.

Table 22 reports selected question-type results at the matched three-call budget. BFR-Text has higher Temporal LLM-Suff@Set; the Open Domain rows report the observed answer scores on eight questions.

Table 22: **Selected question-type results at a matched three-call budget.** LoCoMo, C3, n=200. Values are percentages; differences are BFR-MV minus BFR-Text in percentage points, computed before rounding. Open Domain has eight questions.

Question type	Metric	n	BFR-Text	BFR-MV	Δ
Open Domain	Judge	8	0.0	12.5	+12.5
Temporal	LLM-Suff@Set	50	74.0	70.0	−4.0
Open Domain	F1	8	1.9	3.4	+1.6

On LongMemEval-S the same budget favors BFR-MV on Turn Hit (figure 6). Overall is 95.2 against 92.5 (+2.7 pp) on the 479 items with an answer-turn label, whereas the main tables average Turn Hit over all 500 questions. Preference is +6.7 (n=30) and multi-session is +5.6 (n=125); assistant questions are tied at 100%. Paired outcomes are 13 to 0 (p=2.4×10^{-4}). Multi-session is 7 to 0 (raw p=0.016; Bonferroni over six types p=0.094). Judge does not separate the two policies.

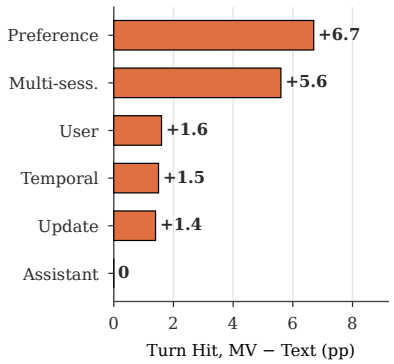


Figure 6: **LongMemEval-S Turn Hit under matched C3 access.** BFR-MV minus BFR-Text, on 479 items with an answer-turn label; differences are percentage points.