TOPOMEM: COMPILING EVIDENCE TOPOLOGY INTO EXECUTABLE MEMORY FOR LONG-CONTEXT REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

Existing approaches to long-context reasoning use direct inference, retrieval, compression, or agentic processing to access and manage long inputs. Our controlled analyses suggest that locating and retaining relevant text can be insufficient for tasks requiring dependency tracking or global aggregation. We propose **TopoMem**, a training-free framework that compiles **evidence topology**—how supporting facts are distributed and related—together with the required composition rule and answer schema into an **executable memory contract**. A restricted state-execution fast path handles supported event sequences; other queries use a hybrid compiler that reconciles query semantics with observable context structure to select a typed reducer or model-backed reader. The operator library spans 13 evaluated execution routes, including set construction, dependency closure, global counting, numerical execution, and semantic synthesis. With a frozen Qwen3.5-9B backbone across five benchmarks, TopoMem matches or exceeds BM25, ReadAgent-P, and LLM×MapReduce-v1 in all completed benchmark-level comparisons. Relative to the best-performing of these evaluated baselines on each benchmark, it improves scores by 14.31 points on LongBench-v2, 21.02 task-balanced points on InfiniteBench, and 33.80 points on BABILong. Gains over direct inference (Direct-HT) persist on the full-context-fit subsets of LongBench-v2 and InfiniteBench, although Direct-HT remains 1.11 points stronger on RULER. Routing ablations on a controlled RULER subset favor joint semantic–structural selection over either signal alone. These findings support constructing long-context memory around **query-dependent computation** rather than text retention alone.

1 INTRODUCTION

Long-context reasoning requires both access to supporting evidence and a way to combine it into the requested answer. A question may ask for one fact hidden in a book, every value associated with an entity, a transitive dependency, or a statistic over the entire input. Such queries can share vocabulary while requiring different intermediate states and composition rules. Larger context windows expose more evidence but do not determine which computation preserves it Liu et al. (2024); Zhang et al. (2024a); Kuratov et al. (2024). We study how to choose that memory computation without training the backbone.

Existing systems improve evidence access and retention in complementary ways. Retrieval selects windows, compression preserves a compact synopsis, and agentic systems coordinate reading through sequential workers Zhang et al. (2024b), leader–member teams Zhao et al. (2024), or map–reduce protocols Zhou et al. (2025). These methods can support reasoning, but the retained state may not match a query's composition requirement. For example, if two chunks count items as $A : 5, B : 4$ and $C : 5, B : 4$, retaining only each chunk's winner discards B, the global winner with eight occurrences. The design question is what intermediate state memory must retain for the requested computation to remain possible.

We answer this question with an execution contract. *Evidence topology* τ describes how supporting facts are distributed and related; a composition rule ρ specifies how to combine them, and an

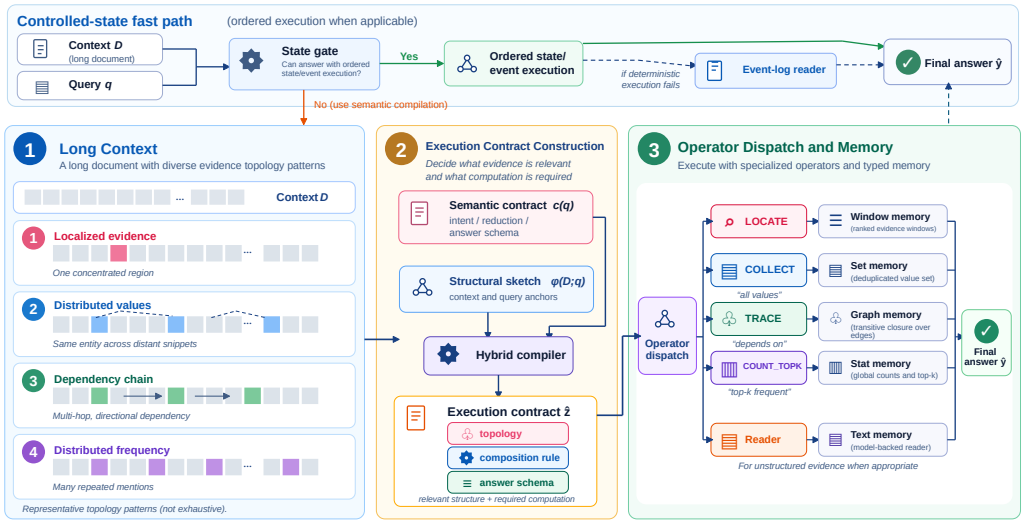


Figure 1: TopoMem overview with four illustrative evidence topologies. The controlled-state path consumes (q, D) and may answer before semantic compilation. Otherwise, the query-only contract $c(q)$ and a query-conditioned structural sketch $\phi(D; q)$ independently enter the hybrid compiler to select a typed operator or model-backed reader.

answer schema $\mathcal{Y}$ specifies the required output. Together they form $z = (\tau, \rho, \mathcal{Y})$. Evidence topology denotes observable evidence patterns rather than a topological invariant or learned graph representation. Localized evidence supports ranked retrieval, distributed relation values support set construction, dependencies support graph closure, and frequencies support global counting before top-k selection. Each pattern calls for a different state to be retained.

We instantiate this idea in **TopoMem**, a training-free hybrid dispatcher with a typed operator library. A restricted controlled-event gate first attempts deterministic state execution. Otherwise, a query-only semantic contract is reconciled with context sketches and explicit structural rules to select either a deterministic reducer or a model-backed reader. The library supports set, graph, frequency, numeric, arithmetic, function, and state operations, together with sparse retrieval, option-wise evidence adjudication, and semantic synthesis. This design preserves exact structure when a pattern is recognized while retaining neural readers for unstructured evidence. The evaluated system uses handwritten grammars and branch-specific fallbacks for supported patterns.

Across five Qwen3.5-9B suites (10,574 examples), TopoMem gains over the strongest completed BM25, ReadAgent-P, or LLM×MapReduce-v1 baseline on LongBench-v2, InfiniteBench, and BABILong (Table 1). Direct-HT remains 1.11 points stronger on RULER. The gains persist on full-context-fit inputs, and controlled RULER routing ablations favor the complete policy over restricted variants.

Our contributions are as follows.

- We formulate evidence topology as the structural component of an execution contract that connects evidence organization, compositional reduction, and answer schema.
- We propose TopoMem, which combines semantic compilation and structural rules with typed deterministic reducers and model-backed readers, including an explicit controlled-state execution path.
- We characterize where query-dependent memory helps: gains persist on full-context-fit inputs, a controlled RULER comparison favors the complete routing policy, and task-level analyses reveal concentrated execution benefits and persistent failure cases.

2 Related Work

Long-context evaluation. Needle-in-a-Haystack tests Kamradt (2023) measure whether a model can locate a planted fact as context length and depth vary. RULER adds multi-needle, tracing, and aggregation tasks Hsieh et al. (2024); InfiniteBench mixes retrieval with long-book, dialogue, code, and mathematical reasoning Zhang et al. (2024a); LongBench-v2 tests realistic long-context multiple-choice reasoning Bai et al. (2025); and BabiLong embeds multi-hop relational questions in long distractor contexts Kuratov et al. (2024). Together, these suites distinguish access to evidence from the computation needed to combine it, which motivates our evaluation across both kinds of task.

Evidence access and compression. Retrieval-augmented generation selects supporting passages for a reader Lewis et al. (2020). LongLLMLingua compresses a prompt before generation Jiang et al. (2024), while ReadAgent and its ReadAgent-P variant use page gists to choose passages for closer reading Lee et al. (2024). These methods change what text the reader receives. TopoMem retains neural retrieval when it fits the query, while also selecting typed operations for answers that require set construction, dependency tracing, or aggregation.

Inter-chunk coordination. Chain-of-Agents passes a running state through sequential workers Zhang et al. (2024b); LongAgent gathers evidence through leader–member collaboration Zhao et al. (2024); and LLM×MapReduce-v1 defines structured inter-chunk communication Zhou et al. (2025). Sculptor offers active tools for fragmenting, searching, and summarizing context Li et al. (2026). These systems organize how chunks are accessed and combined. TopoMem makes the composition rule itself a routing decision: a query can invoke graph closure or global counting rather than rely on a generic chunk-level summary.

Structured memory and executable operations. HGMEM organizes retrieved facts in a hypergraph for multi-step reasoning Zhou et al. (2026), while MEMORA separates abstractions, memory values, and retrieval cues Xia et al. (2026). These works show how memory representation affects access to related evidence. TopoMem focuses on the query-time choice of an executable reduction and answer schema, ranging from set union and graph traversal to neural reading.

3 Method

3.1 Problem Formulation and System Overview

Given a long context D and query q, the system must produce an answer y while preserving the evidence structure and composition rule that y requires. We use evidence topology τ for the organization of supporting facts, and combine it with composition ρ and answer schema $\mathcal{Y}$ in the execution contract $z = (\tau, \rho, \mathcal{Y})$. This factorization matters because lexically similar queries can require incompatible computations: a scalar lookup, a set of relation values, a dependency closure, and a global top-k ranking should not share the same memory invariant.

For queries not resolved by the controlled-state fast path, TopoMem compiles this contract in two steps. A query-only semantic contract $c(q)$ proposes the requested intent, composition, and answer schema, while a reference-answer-free structural sketch $\phi(D; q)$ uses query anchors to expose observable evidence structure. The sketch depends on q but not on the model-generated $c(q)$. The hybrid compiler reconciles these signals and dispatches to an operator library $\mathcal{O}$:

$$\hat{z} = g(c(q), \phi(D; q)), \qquad \hat{o} = \pi(\hat{z}), \qquad \hat{y} = \mathrm{Exec}_{\hat{o}}(D, q). \tag{1}$$

An operator either returns $\hat{y}$ directly or constructs a typed memory for a model-backed reader. Here, g and π combine a model-generated contract with deterministic sketches and structural rules to implement the execution choice.

From contract to memory. For a supported top-k query, the contract pairs distributed item evidence (τ), count-then-rank (ρ), and an ordered-list answer ($\mathcal{Y}$). The query probe requests ranking while $\phi(D; q)$ detects numbered items. COUNTTOPK accumulates global counts before ranking; a locally losing item can still win globally. Dependency queries instead retain assignment edges for TRACE to compute closure.

Figure 1 summarizes the execution path. A controlled-state gate first attempts ordered event execution. Remaining queries receive a semantic contract and a query-conditioned structural sketch; the compiler chooses a deterministic reduction or prepares evidence for a neural reader. The selected branch determines abstention and fallback behavior. The figure illustrates four representative topologies, and Sections 3.2–3.5 describe their execution. Appendix G.12 reports the complete observed route inventory.

3.2 CONTROLLED-STATE EXECUTION FOR ORDERED EVENTS

State questions require executing ordered changes rather than retrieving an isolated fact. A restricted controlled-state gate identifies location, possession, transfer, and spatial-event questions and extracts an ordered event sequence from D. The executor maintains state and history, returning a supported answer before semantic compilation. If execution cannot resolve an eligible query, a model-backed reader uses the event log; questions outside the gate continue to the hybrid dispatcher. The gate's grammar and event handling are specified in Appendix D.1.

3.3 SEMANTIC CONTRACTS AND STRUCTURAL REFINEMENTS

The query specifies the requested operation and answer format, while observable context patterns indicate which evidence structures are available. TopoMem uses these complementary constraints to select a memory computation for queries not resolved by the controlled-state path. A query-only Qwen3.5-9B probe proposes a semantic contract $c(q)$ with intent, answer schema, cardinality, and reduction. Independently of the model-generated contract, a reference-answer-free sketch $\phi(D; q)$ uses query anchors to expose assignments, numeric streams, anchor-local values, and ordering cues. The sketch depends on q; the compiler reconciles it with $c(q)$ before selecting an operator.

Structural gates select typed routes when requested computation and observed context patterns are compatible. Specialized rules handle supported function definitions, arithmetic streams, and closed-choice inputs; other queries use sparse or multi-evidence retrieval. These rules use neither reference answers nor benchmark identifiers. Their restricted lexical and grammatical patterns define the evaluated scope. Appendix D.1 specifies contract parsing, gates, and overrides.

3.4 TYPED OPERATOR LIBRARY

The operators differ in the intermediate state they retain for downstream computation. COLLECT constructs a deduplicated relation-value set; TRACE preserves assignment dependencies for transitive resolution; and COUNTTOPK accumulates full-context item counts before global ranking. REDUCE, SLICEEXECUTE, and JOIN execute supported numeric aggregates, arithmetic streams, and function dependencies.

Model-backed paths retain evidence for interpretation. Sparse and multi-evidence lookup rank context windows, restore their source order, and ask a model to answer from the selected text. For closed-choice questions, VERIFYSELECT builds model-generated support, contradiction, or unknown annotations for each option, then adjudicates against the ledger and retrieved evidence. ORDER arranges temporal or indexed evidence; FUSE and COMPRESS build rolling semantic memories before a final answer call. Thus selecting a typed route does not necessarily remove neural generation. Appendix D.1 gives parsing scope, budgets, and operator behavior; Appendix G.12 lists the routes observed in the main evaluation.

3.5 BRANCH-SPECIFIC EXECUTION AND FALLBACK

Execution and failure handling depend on the selected branch. Some reducers abstain and defer to sparse retrieval when they cannot produce an answer; the state path can instead pass its event log to a reader. A deterministic state answer needs no model call, while model-backed paths may

compile or read more than once. Appendix D.1 documents branch-specific fallbacks and repeated calls; Appendix H.4 reports the available computation records and their accounting limits.

4 EXPERIMENTS

The evaluation asks where gains occur, whether they persist when direct prompts fit the context window, and how the complete router compares with restricted variants. Task slices, fit-only controls, and routing interventions address these questions separately.

4.1 EXPERIMENTAL SETUP

Benchmarks and scoring. We use Qwen3.5-9B with greedy decoding on five suites: 225 classic Needle-in-a-Haystack (NIAH) cases, all 503 LongBench-v2 questions, 3,946 InfiniteBench examples from 12 tasks, 3,900 RULER examples from 13 tasks and three context scales, and 2,000 BABILong examples covering QA1–QA5 at four lengths. Official scorers retain inference failures as zeros; preliminary and determinism checks do not enter the main scores.

Baselines. We compare with BM25 retrieval, LLM×MapReduce-v1, ReadAgent-P, and one-call direct long-context inference (Direct-HT). Compatible runs share the backbone, manifests, answer limits, decoding, and official scorers. Direct-HT uses the full prompt when it fits the 262,144-token window and a fixed head–tail truncation otherwise. Appendix D records incomplete cells, backend exceptions, and protocol details.

4.2 OVERALL RESULTS

Benchmark-level performance. Table 1 summarizes the main benchmark results. TopoMem matches or exceeds BM25, ReadAgent-P, and LLM×MapReduce-v1 in each reported benchmark comparison. Against Direct-HT, it gains 12.52 points on LongBench-v2, 16.31 task-balanced points on InfiniteBench, and 33.15 points on BABILong, ties on NIAH, and trails by 1.11 points on RULER. Section 4.3 examines the task-level variation.

Table 1: Main benchmark results with Qwen3.5-9B (%; higher is better). InfiniteBench reports the task-balanced mean with the micro mean in parentheses. Dashes mark incomplete cells; inference failures remain zero-scored. Bold marks the best result in each column. ReadAgent-P results and BM25 BABILong use cross-backend inference (Appendix D).

Method	NIAH ($n = 225$)	LongBench-v2 ($n = 503$)	InfiniteBench ($n = 3{,}946$)	RULER ($n = 3{,}900$)	BABILong ($n = 2{,}000$)
Direct-HT	**100.00**	31.61	47.95 (60.78)	**96.15**	47.25
BM25	**100.00**	29.82	43.24 (57.52)	86.41	31.75
ReadAgent-P	21.78	5.57	39.35 (52.87)	82.76	9.90
LLM×MapReduce-v1	**100.00**	27.44	–	–	46.60
TopoMem	**100.00**	**44.14**	**64.26** (**71.32**)	95.04	**80.40**

Other baseline readers. Among BM25, ReadAgent-P, and LLM×MapReduce-v1, the strongest completed baseline is BM25 on LongBench-v2 and InfiniteBench and LLM×MapReduce-v1 on BABILong. TopoMem gains 14.31 points on LongBench-v2 (95% CI [9.15, 19.48]), 21.02 task-balanced points on InfiniteBench ([19.71, 22.32]), and 33.80 points on BABILong ([31.20, 36.40]) against those respective baselines. It also gains 8.64 points over BM25 on RULER ([7.63, 9.64]), gains 16.70 over LLM×MapReduce-v1 on LongBench-v2 ([11.53, 22.07]), and ties it on NIAH. The 48.65-point BABILong gap over BM25 and 24.91-point InfiniteBench gap over ReadAgent-P are descriptive: BM25 BABILong and all ReadAgent-P results use different inference backends from TopoMem (Appendix D).

The gains arise through different paths: supported arithmetic and function execution accounts for most of InfiniteBench's task-balanced improvement, the state fast path covers many BABILong

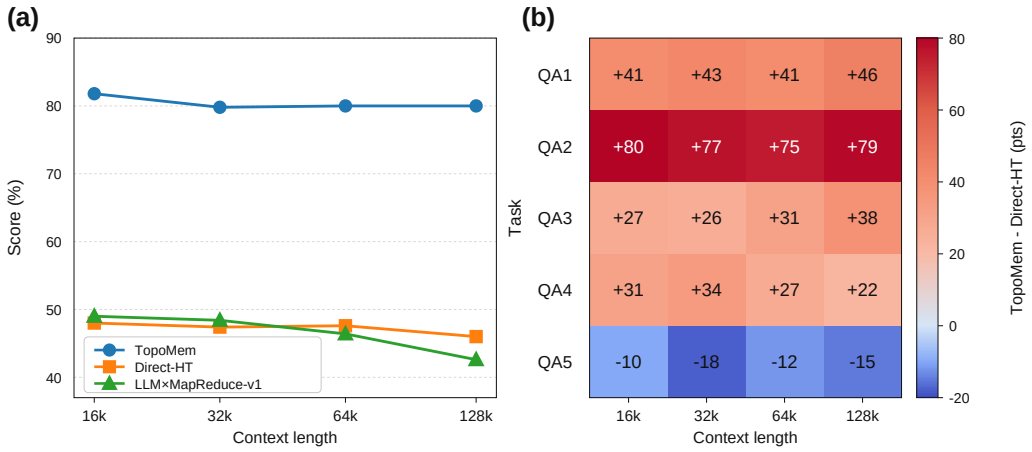


Figure 2: Formal BABILong results across 16k–128k contexts (QA1–QA5; 100 examples per task and length). (a) Mean score for the selected methods TopoMem, Direct-HT, and LLM×MapReduce-v1. (b) TopoMem minus Direct-HT by task and length, in percentage points. TopoMem underperforms Direct-HT on QA5 at every evaluated context length.

cases, and LongBench-v2 uses structured option-wise adjudication. The RULER interventions separately test the hybrid router. Preliminary results for additional baselines appear in Appendix H.1.

4.3 Task-level gains and failure boundaries

InfiniteBench: a concentrated execution benefit. The benchmark-level gain is concentrated in two execution tasks. TopoMem improves on six of twelve InfiniteBench tasks, ties on two, and trails Direct-HT on four (Appendix G). Under equal task weighting, `code_run` (+98.00 points) and `math_calc` (+100.00) contribute 16.50 points to the overall +16.31-point gain; the other ten tasks contribute approximately −0.19. The aggregate therefore reflects an advantage on supported execution tasks rather than uniform improvement across long-context questions.

BABILong: stable length averages, uneven task performance. TopoMem's equal-task mean changes from 81.8% at 16k to 80.0% at 128k, while Direct-HT changes from 48.0% to 46.0%. Yet QA3 remains below ceiling, and QA5 trails Direct-HT at every evaluated length. Figure 2 shows both the length trend and task-level differences; Appendix G reports the detailed task decomposition. Length stability thus coexists with weaker three-supporting-fact and ternary-relation reasoning.

RULER and LongBench-v2: different failure profiles. Across 600 RULER QA1/QA2 examples, TopoMem scores about 68.83% versus Direct-HT's 78.50%; both are near ceiling on the other eleven tasks (99.81% versus 99.36%). LongBench-v2 instead shows positive point differences in all six domains and fourteen of sixteen task categories, with one tie and one decline. Its gains are 1.11, 16.28, and 24.07 points in the short, medium, and long length bins, while the easy–hard score gap narrows from 7.84 to 2.74 points. All 503 LongBench-v2 cases use VerifySelect, so this comparison measures the end-to-end performance of that route.

4.4 Context-window and retrieval controls

Does avoiding truncation explain the gains? We compare TopoMem and Direct-HT on examples whose direct prompts fit the 262K-token context window. TopoMem still gains 9.68 points on 403 LongBench-v2 examples and 10.14 micro-score points on 3,654 InfiniteBench examples (Figure 3). These gains persist when direct inference sees the full prompt. Appendix G.11 reports both strata with paired intervals.

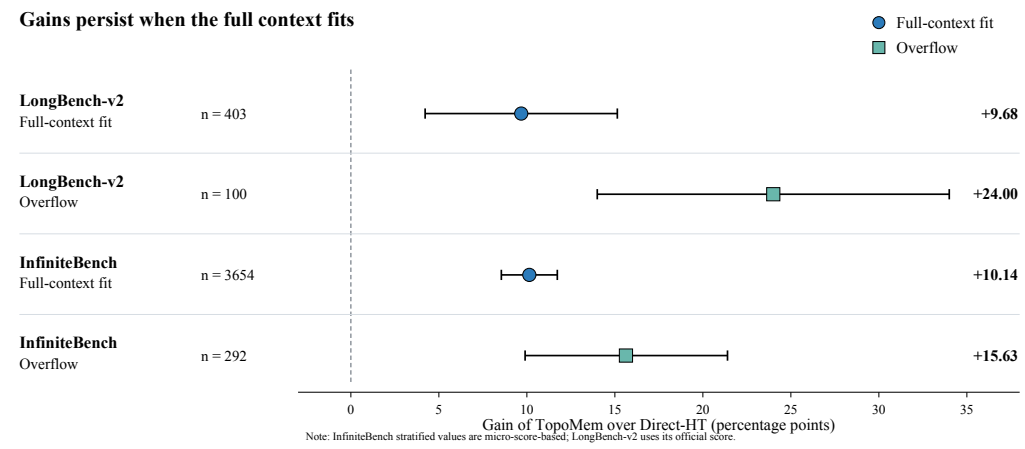


Figure 3: TopoMem minus Direct-HT on full-context-fit and overflow subsets (percentage points; paired 95% bootstrap intervals). LongBench-v2 uses its official score (403 fit, 100 overflow); InfiniteBench uses micro score (3,654 fit, 292 overflow).

More retrieved text is not sufficient. On the same 2,000 BABILong IDs, BM25 top-k=4 scores 34.85%, versus 31.75% at k=8, while using 49.07M rather than 78.61M observed prompt tokens. At a nominal 32k retrieval budget, chunk-size variants score 34.35–34.85%, while neighbor and boundary retention score 33.75% and 33.55%; none approaches TopoMem's 80.40%. BM25 BABILong uses Ascend and is not compute matched (Appendix G.10).

4.5 Routing policy and topology-dependent execution

On 620 shared Ascend examples (520 RULER; 100 LongBench-v2), we compare the full hybrid router with semantic-only, structural-only, and fixed sparse routing. These end-to-end variants change downstream computation as well as route choice; Appendix C defines each arm.

Does the complete routing policy help? On RULER, full hybrid reaches 85.77%, versus 78.52% for fixed sparse routing, 72.12% for semantic-only, and 70.58% for structural-only. Its paired gains are 7.25 points over fixed sparse routing (95% CI [6.58, 7.79]), 13.65 over semantic-only ([13.46, 14.04]), and 15.19 over structural-only ([13.46, 16.92]). On LongBench-v2, full and structural-only tie at 47.00%; the 10-point gain over fixed sparse routing has interval [0, 20]. The complete policy helps on the tested RULER subset, while the LongBench-v2 result follows the closed-choice path retained by structural-only.

Where does topology-dependent execution matter? In the H200-based topology-removal study, the full method gains 14.94 RULER points over the no-topology variant. Among examples assigned to `COUNT_TOPK` and `TRACE` by the full method, the gains are 49.29 and 100.00 points, respectively. These routes account for 81 of 106 favorable per-example transitions; 358 neural-locate cases are unchanged, while `COLLECT` loses 4.17 points. The larger end-to-end differences occur on global aggregation and dependency tracing cases (Table 31). Verification and fixed-budget variants show no accuracy gain on this subset.

The two studies use different inference environments. Their full-method RULER scores differ by 7.50 points despite the same router source and benchmark metrics: 40 `cwe` cases switch from `COUNT_TOPK` on H200 to `REDUCE` on Ascend. The source of this difference remains unresolved, so ablation effects are interpreted within each environment (Appendix C).

4.6 Execution coverage and routing sensitivity

The suites exercise different paths. Main-evaluation predictions exercise 13 routes (Appendix G.12). NIAH uses neural retrieval throughout, and all 503 LongBench-v2 examples use the closed-choice reader. Typed execution covers 722/3,946 InfiniteBench examples (18.3%),

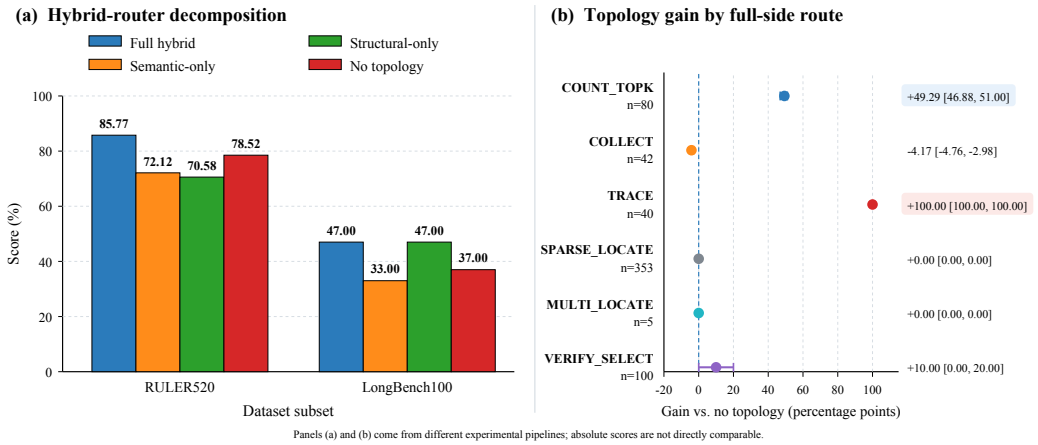


Figure 4: Routing and topology-removal ablations in two inference environments. (a) Four routing variants evaluated on the same 520 RULER and 100 LongBench-v2 examples on Ascend. (b) A separate H200 topology-removal experiment stratified by the full-method route; points show paired TopoMem-minus-no-topology differences with 95% intervals.

1,205/3,900 RULER examples (30.9%), and 1,452/2,000 BABILong examples (72.6%). Across all 10,574 examples, typed, structured-reader, and neural-retrieval routes account for 32.0%, 7.9%, and 60.1%. These counts show which computation families each suite exercises; the BABILong state fast path bypasses semantic compilation.

Controlled phrasing perturbations. On 300 paired question rewrites (240 RULER; 60 LongBench-v2), routes agree in 285 pairs (95.0%), including 225/240 RULER and 60/60 LongBench-v2. Rewriting *what are the* as *name the* accounts for 14 of 15 route changes (31/45 route agreement); the other rewrite families agree on 254/255 pairs. The test measures route stability under these lexical substitutions (Appendix B).

Preliminary and main benchmark breakdowns are reported in Appendices F and G, respectively.

5 Limitations

The main benchmark results use one 9B backbone with greedy decoding. LongLLMLingua and CoA lack main benchmark results. All ReadAgent-P comparisons and BM25 BABILong use a different inference backend from TopoMem. On LongBench-v2, 416 of 494 non-error ReadAgent-P outputs lack a scorable choice label, so its accuracy reflects both output format and answer quality. The 30 preliminary LongBench-v2 examples recur in the 503-question main evaluation; their preliminary scores are excluded from main averages, but they are not a disjoint holdout. Performance is uneven: Direct-HT is 1.11 points stronger on RULER, BABILong QA5 trails its direct baseline, and two execution tasks account for the InfiniteBench aggregate gain.

The routing ablations compare end-to-end policies with different downstream computation and direct paths. They support a topology-dependent execution benefit on tested RULER cases, while separate semantic–structural interaction and accuracy effects of verification or budgeting remain unidentified. Task slices, route strata, and fit/overflow bins describe where differences occur. Fit-only gains rule out truncation as the sole explanation for the LongBench-v2 and InfiniteBench differences, while leaving operator attribution open. Question rewrites measure route stability under lexical substitutions rather than answer robustness. Handwritten grammars and repeated-event deduplication constrain generalization. Main-run call accounting omits semantic compilation, leaving only lower bounds for full-method computation. The 40 `cwe` route changes between H200 and Ascend show sensitivity to inference environment; their cause remains unresolved. Appendices A and D give the detailed boundaries.

6 CONCLUSION

TopoMem uses execution contracts to organize query-dependent memory computation. Across five Qwen3.5-9B suites, the largest task-level gains occur on supported arithmetic, function, and event tasks that use explicit execution; LongBench-v2 uses structured option-wise evidence adjudication. Separate RULER interventions favor the full end-to-end hybrid router over the tested alternatives. For the supported task families, these findings suggest that computation-aware memory can be more effective than retaining relevant text alone.

REPRODUCIBILITY STATEMENT

The available anonymous code supplement contains the evaluated implementation, public-data adapters, scoring code, and model-free tests. Appendix D describes the implementation and evaluation protocol. Full evaluation predictions and paired-analysis scripts are retained separately but are not part of that code package. An exact checkpoint revision and complete software lockfile are also unavailable, limiting independent score recomputation from saved outputs and exact reconstruction of the inference setup.

AI USE STATEMENT

We used generative AI tools as an assistive aid in the preparation of this manuscript, specifically for proofreading, grammar checking, and language refinement to improve clarity and readability. All research ideas, experimental designs, theoretical claims, and final content were conceived, verified, and approved by the authors.

REFERENCES

Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. LongBench v2: Towards deeper understanding and reasoning on realistic long-context multitasks. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3639–3664. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.183. URL https://aclanthology.org/2025.acl-long.183/.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. Ruler: What's the real context size of your long-context language models? In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=kIoBbc76Sy.

Huiqiang Jiang, Qianhui Wu, Xufang Luo, Dongsheng Li, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. LongLLMLingua: Accelerating and enhancing LLMs in long context scenarios via prompt compression. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1658–1677. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.91. URL https://aclanthology.org/2024.acl-long.91/.

Greg Kamradt. LLMTest Needle in a Haystack. https://github.com/gkamradt/LLMTest_NeedleInAHaystack, 2023. GitHub repository.

Yuri Kuratov, Aydar Bulatov, Petr Anokhin, Ivan Rodkin, Dmitry Sorokin, Artyom Sorokin, and Mikhail Burtsev. Babilong: Testing the limits of llms with long context reasoning-in-a-haystack. *Advances in Neural Information Processing Systems*, 37:106519–106554, 2024.

Kuang-Huei Lee, Xinyun Chen, Hiroki Furuta, John Canny, and Ian Fischer. A human-inspired reading agent with gist memory of very long contexts. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 26396–26415. PMLR, 2024. URL https://proceedings.mlr.press/v235/lee24c.html.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33: 9459–9474, 2020.

Mo Li, L. H. Xu, Qitai Tan, Long Ma, Flood Sung, Ting Cao, and Yunxin Liu. Sculptor: Empowering LLMs with cognitive agency via active context management. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=HPeiH7da0Z.

Yangning Li, Shaoshen Chen, Yinghui Li, Yankai Chen, Hai-Tao Zheng, Hui Wang, Wenhao Jiang, and Philip S. Yu. AdmTree: Compressing lengthy context with adaptive semantic trees. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-1348. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/39a397e1721555ddfa83cea2de225760-Abstract-Conference.html.

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

Menglin Xia, Xuchao Zhang, Shantanu Dixit, Paramaguru Harimurugan, Rujia Wang, Victor Rühle, Robert Sim, Chetan Bansal, and Saravan Rajmohan. MEMORA: A harmonic memory representation balancing abstraction and specificity. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026. URL https://arxiv.org/abs/2602.03315.

Xinrong Zhang, Yingfa Chen, Shengding Hu, Zihang Xu, Junhao Chen, Moo Khai Hao, Xu Han, Zhen Leng Thai, Shuo Wang, Zhiyuan Liu, et al. ∞Bench: Extending Long Context Evaluation Beyond 100K Tokens. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15262–15277. Association for Computational Linguistics, 2024a. doi: 10.18653/v1/2024.acl-long.814. URL https://aclanthology.org/2024.acl-long.814/.

Yusen Zhang, Ruoxi Sun, Yanfei Chen, Tomas Pfister, Rui Zhang, and Sercan Ö Arık. Chain of agents: Large language models collaborating on long-context tasks. *Advances in Neural Information Processing Systems*, 37:132208–132237, 2024b.

Jun Zhao, Can Zu, Hao Xu, Yi Lu, Wei He, Yiwen Ding, Tao Gui, Qi Zhang, and Xuanjing Huang. Longagent: Scaling language models to 128k context through multi-agent collaboration. *arXiv preprint arXiv:2402.11550*, 2024.

Chulun Zhou, Chunkang Zhang, Guoxin Yu, Fandong Meng, Jie Zhou, Wai Lam, and Mo Yu. HGMEM: Hypergraph-based working memory to improve multi-step RAG for long-context complex relational modeling. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026. URL https://arxiv.org/abs/2512.23959.

Zihan Zhou, Chong Li, Xinyi Chen, Shuo Wang, Yu Chao, Zhili Li, Haoyu Wang, Qi Shi, Zhixing Tan, Xu Han, Xiaodong Shi, Zhiyuan Liu, and Maosong Sun. LLM×MapReduce: Simplified long-sequence processing using large language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 27664–27678. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.1341. URL https://aclanthology.org/2025.acl-long.1341/.

A Detailed limitations

Comparison scope. The main benchmark comparisons are those reported in Table 1. LLM×MapReduce-v1 has no InfiniteBench or RULER result; LongLLMLingua and CoA were evaluated only on the smaller preliminary sample. ReadAgent-P was evaluated on all five main suites, with five zero-scored InfiniteBench inference errors and nine zero-scored LongBench-v2 context-window failures. Preliminary scores use different samples and cannot replace missing main results. The BABILong BM25 retrieval diagnostics also do not match full-method computation.

Mechanism evidence. Removing topology lowers RULER performance, whereas removing verification leaves RULER unchanged and raises the LongBench-v2 100-example point estimate. Fixed budgeting yields the same predictions as the full method on all 620 ablation examples. The interventions alter downstream computation and, in some variants, direct execution paths; they do not isolate a semantic–structural interaction or a separate accuracy effect of verification. Route-specific scores describe where the end-to-end differences occur. The H200 and Ascend ablations are interpreted separately.

Backbone, backend, and accounting scope. All main experiments use one 9B backbone with greedy decoding. TopoMem predictions were generated on H200, while ReadAgent-P and the BABILong BM25 baseline used Ascend. In a 30-example cross-backend consistency check, BM25 retrieval memories matched in all 30 cases but final predictions matched in only 24. Inference environment is therefore a potential confound for those comparisons. Bootstrap intervals describe finite-example resampling, not variation across models, decoding seeds, or hardware. RULER and BABILong contain synthetic tasks; high aggregate RULER performance masks weaker QA1/QA2 results. The main-run instrumentation omits semantic-compilation calls, so full-method token and efficiency comparisons are unavailable. Direct-HT is 1.11 points higher than TopoMem on RULER, while its head–tail rule truncates 100 LongBench-v2 and 292 InfiniteBench prompts. Task, domain, and length bins reuse the same examples, and fit/overflow intervals do not estimate an interaction.

Rule generalization. The implementation contains hand-engineered event, arithmetic, function, and option grammars. Avoiding benchmark identifiers does not establish benchmark-independent generalization. The controlled question-phrasing perturbation test in Appendix B finds 15 route changes among 300 pairs; it does not test answer stability, paraphrased event syntax, or repeated identical events. The current ablations also do not isolate every extended operator. In particular, the controlled-state parser's global deduplication can discard temporally meaningful recurrence. In a constructed sequence "Mary moved to the kitchen; Mary moved to the hallway; Mary moved to the kitchen," deleting the repeated final sentence changes the state. This is a method counterexample, not a measured benchmark failure; correcting the frozen executor would require a new evaluation.

B Controlled question-phrasing route sensitivity

We evaluated routing stability on 300 paired question rewrites: 240 RULER examples and 60 LongBench-v2 examples selected by label-blind strata from the 620-example ablation subset. Each pair shared the context, answer options, model checkpoint, and greedy decoding, differing only in a deterministic lexical substitution. Both question versions used the unchanged TopoMem router on Ascend without answer generation. All 600 routing evaluations completed without inference errors.

Table 2: Route agreement under lexical question substitutions on Ascend.

Subset	Same route / pairs	Agreement (%)
RULER	225 / 240	93.75
LongBench-v2	60 / 60	100.00
All	285 / 300	95.00

The route changes are concentrated. Fourteen of the 15 changes occur in the RULER `cwe` task (15/29 routes stable), where replacing *what are the* with *name the* changes `REDUCE` to `COUNT_TOPK`; the remaining change is `COLLECT` to `SPARSE_LOCATE`. Table 3 reports all seven rewrite families.

Table 3: Route agreement by question-rewrite family. The substitutions are listed in Table 4.

Rewrite family	Same route / pairs	Agreement (%)
`after`	1 / 1	100.00
`list`	15 / 15	100.00
`what_are_the`	31 / 45	68.89
`what_is_the`	182 / 182	100.00
`when_did`	3 / 3	100.00
`which`	51 / 52	98.08
`which_option`	2 / 2	100.00
All	285 / 300	95.00

Table 4: Literal first-occurrence substitutions used by the seven rewrite families represented in the 300-pair test. Matching is case-insensitive and the first replacement letter follows the matched capitalization. These are replacement spans, not complete questions.

Family	Matched span	Replacement span
`after`	after	subsequent to
`list`	list	enumerate
`what_are_the`	what are the	name the
`what_is_the`	what is the	state the
`when_did`	when did	at what time did
`which`	which	identify which
`which_option`	which option	which alternative

C Additional mechanism ablations

Routing variants. All four variants use the same Qwen3.5-9B checkpoint, answer limits, greedy decoding, and 620 evaluation examples. Full hybrid uses the evaluated TopoMem router. Semantic-only uses the query-only semantic compiler without context-conditioned routing gates or the state, code-fault, and closed-choice direct paths. Structural-only omits the model-generated semantic contract but retains context and question-format rules, including those direct paths; ambiguous structures use sparse retrieval. No topology replaces dispatch with eight BM25-style evidence windows and one answer call, without typed or direct execution paths. The no-verification variant omits the option-wise ledger call and ledger text in the final-choice prompt; it also changes ledger-dependent budgeting. The fixed-budget variant forces eight sparse windows and caps rolling evidence at its first 6,000 tokens without collapse, while retaining the main routing policy. These are end-to-end variants with different downstream computations and direct paths, not a factorial test that isolates a semantic–structural interaction.

Topology-removal study on H200. This study uses 520 RULER examples and a stratified 100-question LongBench-v2 subset. Its full-method predictions come from the main H200 evaluations of 3,900 RULER and 503 LongBench-v2 examples; intervention predictions were generated separately on H200. The four-arm routing study instead reran every variant, including full hybrid, on Ascend. Both studies use the same 620 examples, router source, and official metric implementations, but their inference environments differ.

On RULER, 477/520 full-method predictions match across environments. All 40 `cwe` examples change from `COUNT_TOPK` on H200 to `REDUCE` on Ascend and lose score; one other example favors Ascend. The resulting full-method scores are 93.27% and 85.77%, respectively, a 7.50-point difference. The difference lies in predictions and routes, while the backend-versus-runtime cause remains unresolved. On the 100 LongBench-v2 examples, the scores differ by one point,

with nine cases favoring H200 and eight favoring Ascend. We interpret ablation effects within each environment. On H200, removing topology lowers RULER by 14.94 points (paired 95% CI [11.84, 18.10]) and LongBench-v2 by 10.0 points ([-2.0, 22.0]). Removing verification leaves RULER unchanged and raises the LongBench-v2 point estimate by 5.0 points; fixed budgeting leaves all 620 scores unchanged.

Table 5: Topology-removal and related H200 ablations (%). Parentheses show full TopoMem minus the variant and paired 95% bootstrap intervals in percentage points. These scores are separate from the four-arm Ascend comparison in Table 6.

Variant	RULER520	LongBench100
Full TopoMem	93.27	48.00
No topology	78.33 (+14.94; [11.84, 18.10])	38.00 (+10.00; [-2.00, 22.00])
No verification	93.27 (+0.00; [0, 0])	**53.00** (-5.00; [-16.00, 6.00])
Fixed evidence budget	93.27 (+0.00; [0, 0])	48.00 (+0.00; [0, 0])

The LongBench-v2 +10.00-point topology contrast also appears by the VERIFYSELECT route in Table 31, but its interval there is [0, 20] rather than the [-2, 22] shown above. The H200 subset analysis resamples the 100 paired examples without task strata (10,000 replicates; seed 20260911); route localization resamples within task strata (20,000 replicates; seed 20260919). The point estimate is unchanged because the two intervals use different resampling procedures.

Computational cost of the routing variants. Each Ascend variant covers the same 620 examples and uses the official benchmark metrics. Table 6 reports its model-call and token totals. Paired score intervals resample examples within benchmark, task, and cell strata for 10,000 replicates (seed 20260925). On RULER, full hybrid gains 7.25 points over no topology ([6.58, 7.79]), 13.65 over semantic-only ([13.46, 14.04]), and 15.19 over structural-only ([13.46, 16.92]). On LongBench-v2, it gains 10 points over no topology ([0, 20]) and 14 over semantic-only ([3, 25]), and ties structural-only at 47.00% ([0, 0]). Semantic-only uses 3,465 calls versus 3,096 for full hybrid yet scores lower on RULER; the comparison is not compute matched. A post-hoc unstratified paired bootstrap (10,000 replicates; seed 20260925) on the same 520 RULER examples gives the full-minus-semantic difference as +13.65 points with 95% CI [10.96, 16.44], compared with [13.46, 14.04] under task stratification. The latter conditions on 40 examples per RULER task; both procedures preserve the same point estimate.

Table 6: Complete call and token accounting for the separate four-arm Ascend comparison. Scores use the official benchmark metrics on identical examples.

Variant	RULER520	LongBench100	Calls	Prompt tokens	Completion tokens
Full hybrid	**85.77**	47.00	3,096	12,800,001	187,506
Semantic-only	72.12	33.00	3,465	4,734,044	217,279
Structural-only	70.58	**47.00**	528	11,279,611	97,570
No topology	78.52	37.00	620	3,932,799	13,895

D DETAILED EXPERIMENTAL PROTOCOL

Common evaluation controls. The main benchmark methods use Qwen3.5-9B, greedy decoding, fixed answer limits, and official benchmark metrics when supported by the method. The H200 and Ascend ablation studies use the same metric implementations; their absolute scores differ because their predictions differ. The model family is Qwen/Qwen3.5-9B, but an exact checkpoint revision is unavailable in the retained records. Inference failures remain in the denominator with a score of zero.

A 330-example preliminary evaluation spans all five benchmark families, and a separate 30-example check tests determinism. Neither contributes additional observations to the main scores. The preliminary set is not disjoint: all 30 LongBench-v2 source examples also occur in the 503-question main evaluation. The available records do not establish when grammar and prompt choices were fixed relative to preliminary feedback.

Direct and retrieval baselines. Direct-HT is a non-agentic one-request reference using the checkpoint chat template with thinking disabled. It passes the complete serialized context when the prompt fits the 262,144-token window and otherwise retains equal-size token prefixes and suffixes around a fixed, query-independent truncation marker. All NIAH, RULER, and BABILong prompts fit; 100/503 LongBench-v2 and 292/3,946 InfiniteBench prompts require truncation. BM25 uses the same manifests, but its main BABILong evaluation retrieves eight 8k-token chunks, distinct from TopoMem's overlapping 768-token sparse windows and from the fixed sparse no-topology intervention. BM25 BABILong ran on Ascend TP1 while TopoMem ran on H200; a transport check reproduced 30/30 retrieval memories but 24/30 final predictions. These are same-example rather than backend-equivalent comparisons.

Agentic and compression baselines. LLM×MapReduce-v1 uses the author's V1 algorithm and has main benchmark results for NIAH, LongBench-v2, and BABILong. ReadAgent-P Lee et al. (2024) uses a tokenizer-aware adaptation that preserves pagination, source order, lookup limits, and raw-page expansion. It was evaluated on all five main suites on Ascend, whereas TopoMem used H200. Nine LongBench-v2 context-window failures and five InfiniteBench inference errors remain zero-scored. The output-format diagnostic and same-example comparisons appear in Appendix H.2. LongLLMLingua Jiang et al. (2024) uses its compressor and answer-recovery pipeline; CoA Zhang et al. (2024b) uses sequential workers and a final manager. Both are reported only in the smaller preliminary evaluation (Appendix H.1).

Analysis and interpretation. Task, length, and domain results regroup the saved main-evaluation predictions. Removing the two execution tasks and reaveraging the ten printed InfiniteBench task means yields about 57.11 for TopoMem and 57.34 for Direct-HT. Length, difficulty, and context-window-fit bins describe overlapping examples in different ways; fit/overflow intervals estimate within-stratum differences rather than their interaction. The BM25 retrieval controls use the same BABILong examples but are not compute matched to the full method (Appendix G.10). The routing and topology-removal studies use Ascend and H200, respectively, so ablation effects are interpreted within each environment (Appendix C).

Software and recording scope. The available anonymous code package includes the evaluated method, public-data adapters, scoring code, and model-free tests. Saved per-example predictions and scores support the reported analyses but are not included in that code package. The runtime record identifies a container, PyTorch `2.13.0+cu130`, eight workers, and deterministic PyTorch/cuDNN settings. Exact Python, Transformers, Accelerate, driver, and cuDNN versions and a complete environment lockfile were not retained. The main-run call counter omits semantic-compilation requests; Appendix H.4 therefore reports observed totals as lower bounds.

D.1 TopoMem implementation details

Controlled-state gate and execution. The first gate recognizes restricted English question forms about locations, prior locations, transfers, carrying sets and counts, and spatial relations. Regular expressions extract movement, location, possession, transfer, and spatial facts. Matches are ordered by source position and globally deduplicated by case-insensitive text. The gate normally requires at least two distinct facts; a carrying question can pass with one. The deterministic executor maintains location possibilities, negative facts, location histories, object holders, carrying sets, transfers, and inverse spatial relations. A supported answer returns immediately with zero model calls. If the gate passes but execution returns no answer, a reader receives the ordered event log and makes one answer-generation call. Only an ineligible state gate delegates to semantic compilation. This is a restricted event language: unsupported wording can be missed, and global deduplication can remove a meaningful repeated event.

Semantic contract and structural gates. A greedy Qwen3.5-9B call sees only q and emits a JSON contract with one of eight intents: lookup, collection, dependency tracing, frequency ranking, numeric aggregation, temporal ordering, evidence fusion, or global summarization. Other fields specify a scalar, set, ordered-list, or free-text answer; cardinality k; and a reduction of none, sum, mean, minimum, maximum, or count. The prompt asks the compiler to ignore explicitly negated distractor cues. Malformed or out-of-vocabulary contracts map to an unknown intent. The parser

clamps k to $[1, 20]$ and projects selected inconsistencies: an explicit reduction selects numeric aggregation, a set-valued lookup selects collection, and a multi-item ordered-list lookup selects temporal ordering.

The system extracts content-bearing query anchors and computes $\phi(D; q)$ from assignments, numeric streams, anchor-local values, dates, sequence markers, and estimated context extent. Base gates require compatible intent and structure. Tracing requires at least two assignment edges and a numeric query target; ranking requires $k > 1$ and at least ten numbered items or ellipsis markers; collection requires at least two anchor-containing clauses with numeric values. Numeric aggregation, ordering, fusion, and summarization have separate structural gates. Base selection defaults to lookup when no gate applies. A set-valued collection contract can be refined to tracing when lookup was selected but a parseable assignment graph and explicit numeric target are present. Specialized overrides cover restricted function definitions, sequential arithmetic streams, and closed-choice inputs. Fixed query cues direct remaining lookup cases to sparse or multi-evidence retrieval. These rules consume neither reference answers nor benchmark identifiers; their lexical and grammatical specialization is part of the frozen method.

Retrieval and structured readers. The lookup implementation ranks 768-token windows with 96-token overlap by a BM25-style score, selects eight windows (twelve for multi-evidence lookup), restores source order, and calls the answer model once. Closed-choice VERIFYSELECT selects up to eight broad chunks, normally 8,000 tokens each, then uses two generation stages: an option-wise evidence ledger and answer adjudication against both ledger and retrieved evidence. The model-generated ledger marks support, contradiction, or unknown evidence for each option.

Reducer and memory behavior. COLLECT chooses the rarest distinctive query anchor found in D, selects clauses containing it, extracts relation-local numeric values, and deduplicates values in first-occurrence order. With no anchor or value, it abstains and falls back to LOCATE, rather than returning an empty set as a confident answer. TRACE parses common assignment relations into directed edges, resolves nodes through the graph, and returns all nodes whose transitive resolution reaches the target; a visited set prevents cycles. COUNTTOPK parses numbered item streams or punctuation-separated symbol streams, normalizes and counts items globally, then returns the requested top-k; numbered items may contain multiple words. This avoids composing chunk-level winners into a global ranking. REDUCE extracts numeric values from anchor-local clauses when available, otherwise from the global stream, and applies sum, mean, minimum, maximum, or count. SLICEEXECUTE accumulates intermediate results in restricted addition/subtraction streams. JOIN recursively evaluates named unary-function dependencies with supported constant addition, subtraction, or multiplication, rejecting missing, cyclic, or unsupported definitions. ORDER sorts dated or indexed evidence before a model answer. FUSE and COMPRESS build rolling semantic memory in precision-oriented and comprehensive modes, respectively, before a final answer call. These parsers and reducers cover the specified arithmetic and function grammars.

Fallback and call accounting. The base set, graph, frequency, numeric, and ordered-memory paths use sparse retrieval when they produce no prediction. Direct-return arithmetic and function branches do not implement that same missing-result fallback; the state branch uses the ordered-log reader described above. Output type and cardinality are not universally reverified. Some delegated fallback paths repeat query-only semantic compilation without caching those calls. Sparse answering, option-ledger adjudication, ordered-memory answering, and rolling synthesis add different generation work, so no global one-compiler-call plus one-answer-call bound applies. The ablation runs use a complete model-call observer, but the original full-run instrumentation omits semantic-compilation calls. Appendix H.4 reports observed totals and the resulting lower-bound interpretation.

E DETAILED BENCHMARK BREAKDOWNS

The tables below regroup the same preliminary and main-evaluation examples by task, domain, length, or difficulty. These are different views of the same populations, not additional independent samples. Scores are percentages from the official per-example metrics; inference failures remain

zero-scored. LLMx denotes LLM×MapReduce-v1, LLML denotes LongLLMLingua, and RA-P denotes ReadAgent-P.

The preliminary evaluation contains 330 examples: 25 NIAH, 30 LongBench-v2, 60 InfiniteBench, 195 RULER, and 20 BABILong. The main evaluation contains 225, 503, 3,946, 3,900, and 2,000 examples, respectively. Preliminary BABILong spans QA1–QA10 at 32k/128k; main BABILong spans QA1–QA5 at 16k/32k/64k/128k. The preliminary ReadAgent-P row combines Ascend and H200 inference, whereas its main results use Ascend; CoA's preliminary results use CUDA. TopoMem's main predictions use H200. A dash means that a main benchmark result is not reported, rather than a score of zero. Direct-HT uses one request with the full serialized context when it fits the 262,144-token window and a symmetric token prefix/suffix otherwise.

BABILong task meanings. Following the official benchmark task definitions, QA1, QA2, and QA3 require one, two, and three supporting facts, respectively. QA4 tests binary relations and QA5 ternary relations; argument count is not supporting-fact count. The preliminary-only tasks cover yes/no decisions (QA6), counting (QA7), set enumeration (QA8), negation (QA9), and uncertain knowledge (QA10).

F PRELIMINARY EVALUATION BREAKDOWNS

F.1 PRELIMINARY: CLASSIC NIAH BY LENGTH AND DEPTH

Table 7: Preliminary classic NIAH by length and depth: official score (%). Each cell contains one example; these binary outcomes are not stable cell-level estimates.

Length	Depth (percent)	n	LLMx	LLML	CoA	RA-P	TopoMem
1000	0	1	100.00	100.00	100.00	100.00	100.00
1000	25	1	100.00	100.00	100.00	100.00	100.00
1000	50	1	100.00	100.00	100.00	100.00	100.00
1000	75	1	100.00	100.00	100.00	100.00	100.00
1000	100	1	100.00	100.00	100.00	100.00	100.00
32000	0	1	100.00	100.00	0.00	100.00	100.00
32000	25	1	100.00	0.00	100.00	0.00	100.00
32000	50	1	100.00	100.00	100.00	0.00	100.00
32000	75	1	100.00	100.00	100.00	0.00	100.00
32000	100	1	100.00	100.00	0.00	100.00	100.00
64000	0	1	100.00	100.00	0.00	100.00	100.00
64000	25	1	100.00	0.00	0.00	0.00	100.00
64000	50	1	100.00	100.00	0.00	100.00	100.00
64000	75	1	100.00	0.00	0.00	0.00	100.00
64000	100	1	100.00	100.00	0.00	100.00	100.00
96000	0	1	100.00	100.00	0.00	100.00	100.00
96000	25	1	100.00	100.00	0.00	0.00	100.00
96000	50	1	100.00	0.00	0.00	0.00	100.00
96000	75	1	100.00	0.00	100.00	0.00	100.00
96000	100	1	100.00	0.00	100.00	100.00	100.00
128000	0	1	100.00	100.00	0.00	100.00	100.00
128000	25	1	100.00	100.00	0.00	100.00	100.00
128000	50	1	100.00	100.00	0.00	0.00	100.00
128000	75	1	100.00	0.00	100.00	0.00	100.00
128000	100	1	100.00	100.00	0.00	0.00	100.00

F.2 PRELIMINARY: LONGBENCH-V2 BY DOMAIN

Table 8: Preliminary LongBench-v2 by domain: official score (%). Accuracy by the indicated category.

Domain	n	LLMx	LLML	CoA	RA-P	TopoMem
Code Repository Understanding	5	60.00	40.00	40.00	0.00	40.00
Long In-context Learning	5	60.00	0.00	60.00	20.00	80.00
Long Structured Data Understanding	5	20.00	20.00	0.00	20.00	40.00
Long-dialogue History Understanding	5	20.00	20.00	40.00	40.00	40.00
Multi-Document QA	5	40.00	40.00	60.00	0.00	60.00
Single-Document QA	5	20.00	0.00	60.00	0.00	40.00

F.3 PRELIMINARY: LONGBENCH-V2 BY TASK

Table 9: Preliminary LongBench-v2 by task: official score (%). Accuracy by the indicated category.

Task	n	LLMx	LLML	CoA	RA-P	TopoMem
Academic	9	33.33	11.11	55.56	0.00	44.44
Agent history QA	1	0.00	100.00	0.00	0.00	100.00
Code repo QA	5	60.00	40.00	40.00	0.00	40.00
Dialogue history QA	4	25.00	0.00	50.00	50.00	25.00
Legal	1	0.00	100.00	100.00	0.00	100.00
Table QA	5	20.00	20.00	0.00	20.00	40.00
User guide QA	5	60.00	0.00	60.00	20.00	80.00

F.4 PRELIMINARY: LONGBENCH-V2 BY DIFFICULTY

Table 10: Preliminary LongBench-v2 by difficulty: official score (%). Accuracy by the indicated category.

Difficulty	n	LLMx	LLML	CoA	RA-P	TopoMem
easy	13	53.85	15.38	46.15	15.38	46.15
hard	17	23.53	23.53	41.18	11.76	52.94

F.5 PRELIMINARY: LONGBENCH-V2 BY LENGTH

Table 11: Preliminary LongBench-v2 by length: official score (%). Accuracy by the indicated category.

Length	n	LLMx	LLML	CoA	RA-P	TopoMem
long	5	40.00	0.00	40.00	0.00	40.00
medium	12	41.67	16.67	50.00	33.33	50.00
short	13	30.77	30.77	38.46	0.00	53.85

F.6 PRELIMINARY: INFINITEBENCH BY TASK

Table 12: Preliminary InfiniteBench by task: official score (%). Task-specific official metrics retain their original definitions; scores are not uniformly exact-match accuracy. The task-balanced mean weights the 12 task means equally; the micro mean weights examples equally.

Task	n	LLMx	LLML	CoA	RA-P	TopoMem
code_debug	5	20.00	40.00	0.00	20.00	40.00
code_run	5	0.00	20.00	0.00	0.00	100.00
kv_retrieval	5	100.00	0.00	20.00	40.00	100.00
longbook_choice_eng	5	40.00	60.00	0.00	40.00	60.00
longbook_qa_chn	5	2.44	13.40	18.52	16.18	53.33
longbook_qa_eng	5	38.14	24.51	44.44	23.11	41.82
longbook_sum_eng	5	26.79	19.50	14.09	12.14	26.38
longdialogue_qa_eng	5	40.00	40.00	0.00	20.00	40.00
math_calc	5	0.00	0.00	0.00	0.00	100.00
math_find	5	100.00	60.00	100.00	80.00	100.00
number_string	5	100.00	100.00	100.00	100.00	100.00
passkey	5	100.00	100.00	100.00	100.00	100.00

F.7 PRELIMINARY: RULER BY TASK AND LENGTH

Table 13: Preliminary RULER by task and length: official score (%). All 13 tasks and all three evaluated scales are shown.

Task	Length	n	LLMx	LLML	CoA	RA-P	TopoMem
cwe	8192	5	0.00	16.00	76.00	100.00	100.00
cwe	32768	5	16.00	0.00	44.00	100.00	100.00
cwe	131072	5	20.00	10.00	0.00	94.00	100.00
fwe	8192	5	100.00	86.67	100.00	93.33	100.00
fwe	32768	5	100.00	66.67	100.00	93.33	100.00
fwe	131072	5	100.00	80.00	100.00	100.00	100.00
niah_multikey_1	8192	5	100.00	20.00	100.00	100.00	100.00
niah_multikey_1	32768	5	100.00	0.00	0.00	100.00	100.00
niah_multikey_1	131072	5	100.00	0.00	20.00	100.00	100.00
niah_multikey_2	8192	5	100.00	100.00	100.00	0.00	100.00
niah_multikey_2	32768	5	100.00	100.00	40.00	0.00	100.00
niah_multikey_2	131072	5	100.00	100.00	40.00	60.00	100.00
niah_multikey_3	8192	5	100.00	100.00	80.00	40.00	100.00
niah_multikey_3	32768	5	100.00	100.00	40.00	20.00	100.00
niah_multikey_3	131072	5	100.00	100.00	40.00	0.00	100.00
niah_multiquery	8192	5	90.00	0.00	100.00	100.00	100.00
niah_multiquery	32768	5	80.00	15.00	40.00	100.00	100.00
niah_multiquery	131072	5	70.00	15.00	10.00	100.00	100.00
niah_multivalue	8192	5	90.00	0.00	85.00	100.00	100.00
niah_multivalue	32768	5	90.00	25.00	20.00	100.00	100.00
niah_multivalue	131072	5	60.00	0.00	5.00	100.00	100.00
niah_single_1	8192	5	100.00	100.00	100.00	100.00	100.00
niah_single_1	32768	5	100.00	100.00	80.00	100.00	100.00
niah_single_1	131072	5	100.00	100.00	100.00	100.00	100.00
niah_single_2	8192	5	100.00	40.00	100.00	100.00	100.00
niah_single_2	32768	5	100.00	20.00	80.00	100.00	100.00
niah_single_2	131072	5	100.00	0.00	0.00	100.00	100.00
niah_single_3	8192	5	100.00	20.00	60.00	100.00	100.00
niah_single_3	32768	5	100.00	20.00	60.00	100.00	100.00
niah_single_3	131072	5	100.00	40.00	0.00	100.00	100.00
qa_1	8192	5	60.00	80.00	80.00	80.00	100.00
qa_1	32768	5	100.00	100.00	60.00	80.00	80.00
qa_1	131072	5	80.00	100.00	40.00	40.00	80.00
qa_2	8192	5	80.00	80.00	80.00	60.00	80.00
qa_2	32768	5	40.00	60.00	40.00	40.00	40.00
qa_2	131072	5	20.00	40.00	0.00	60.00	60.00
vt	8192	5	32.00	8.00	100.00	80.00	100.00
vt	32768	5	4.00	4.00	40.00	100.00	100.00
vt	131072	5	12.00	8.00	4.00	100.00	100.00

F.8 PRELIMINARY: BABILONG BY TASK AND LENGTH

Table 14: Preliminary BABILong by task and length: official score (%). Task identifiers follow the benchmark. Preliminary cells have one example; main cells have 100 examples.

Task	Length	n	LLMx	LLML	CoA	RA-P	TopoMem
qa1	32k	1	100.00	0.00	0.00	0.00	100.00
qa1	128k	1	0.00	0.00	0.00	0.00	100.00
qa2	32k	1	0.00	0.00	0.00	0.00	100.00
qa2	128k	1	0.00	0.00	0.00	0.00	100.00
qa3	32k	1	0.00	0.00	0.00	0.00	100.00
qa3	128k	1	0.00	0.00	0.00	0.00	100.00
qa4	32k	1	100.00	0.00	0.00	0.00	100.00
qa4	128k	1	100.00	0.00	0.00	0.00	100.00
qa5	32k	1	100.00	100.00	0.00	100.00	100.00
qa5	128k	1	100.00	100.00	0.00	0.00	100.00
qa6	32k	1	100.00	100.00	0.00	0.00	100.00
qa6	128k	1	0.00	0.00	0.00	0.00	100.00
qa7	32k	1	0.00	0.00	0.00	0.00	100.00
qa7	128k	1	0.00	0.00	0.00	0.00	100.00
qa8	32k	1	0.00	0.00	0.00	0.00	100.00
qa8	128k	1	0.00	0.00	0.00	0.00	100.00
qa9	32k	1	100.00	100.00	100.00	100.00	100.00
qa9	128k	1	100.00	100.00	100.00	0.00	100.00
qa10	32k	1	0.00	100.00	0.00	0.00	100.00
qa10	128k	1	100.00	0.00	0.00	0.00	100.00

G MAIN BENCHMARK BREAKDOWNS

G.1 MAIN: CLASSIC NIAH BY LENGTH

Table 15: Main classic NIAH by length: official score (%). Each length contains all 15 evaluated depths.

Length	n	BM25	LLMx	RA-P	Direct-HT	TopoMem
1000	15	100.00	100.00	100.00	100.00	100.00
10071	15	100.00	100.00	13.33	100.00	100.00
19143	15	100.00	100.00	26.67	100.00	100.00
28214	15	100.00	100.00	20.00	100.00	100.00
37286	15	100.00	100.00	13.33	100.00	100.00
46357	15	100.00	100.00	13.33	100.00	100.00
55429	15	100.00	100.00	13.33	100.00	100.00
64500	15	100.00	100.00	20.00	100.00	100.00
73571	15	100.00	100.00	20.00	100.00	100.00
82643	15	100.00	100.00	6.67	100.00	100.00
91714	15	100.00	100.00	20.00	100.00	100.00
100786	15	100.00	100.00	20.00	100.00	100.00
109857	15	100.00	100.00	13.33	100.00	100.00
118929	15	100.00	100.00	20.00	100.00	100.00
128000	15	100.00	100.00	6.67	100.00	100.00

G.2 MAIN: CLASSIC NIAH BY DEPTH

Table 16: Main classic NIAH by depth: official score (%). Each depth contains all 15 lengths. BM25, LLMx, Direct-HT, and TopoMem score 100 on every individual cell. RA-P scores 49/225 overall; its marginals summarize, rather than eliminate, length–depth variation.

Depth (percent)	n	BM25	LLMx	RA-P	Direct-HT	TopoMem
0	15	100.00	100.00	100.00	100.00	100.00
7	15	100.00	100.00	13.33	100.00	100.00
14	15	100.00	100.00	13.33	100.00	100.00
21	15	100.00	100.00	20.00	100.00	100.00
29	15	100.00	100.00	13.33	100.00	100.00
36	15	100.00	100.00	6.67	100.00	100.00
43	15	100.00	100.00	13.33	100.00	100.00
50	15	100.00	100.00	13.33	100.00	100.00
57	15	100.00	100.00	13.33	100.00	100.00
64	15	100.00	100.00	6.67	100.00	100.00
71	15	100.00	100.00	13.33	100.00	100.00
79	15	100.00	100.00	20.00	100.00	100.00
86	15	100.00	100.00	13.33	100.00	100.00
93	15	100.00	100.00	13.33	100.00	100.00
100	15	100.00	100.00	53.33	100.00	100.00

G.3 MAIN: LONGBENCH-V2 BY DOMAIN

Table 17: Main LongBench-v2 by domain: official score (%). Accuracy by the indicated category.

Domain	n	BM25	LLMx	RA-P	Direct-HT	TopoMem
Code Repository Understanding	50	32.00	40.00	4.00	14.00	48.00
Long In-context Learning	81	28.40	18.52	18.52	34.57	44.44
Long Structured Data Understanding	33	27.27	24.24	3.03	33.33	42.42
Long-dialogue History Understanding	39	33.33	23.08	7.69	33.33	46.15
Multi-Document QA	125	31.20	26.40	0.80	28.80	40.00
Single-Document QA	175	28.57	30.29	3.43	36.57	45.71

G.4 MAIN: LONGBENCH-V2 BY TASK

Table 18: Main LongBench-v2 by task: official score (%). Accuracy by the indicated category.

Task	n	BM25	LLMx	RA-P	Direct-HT	TopoMem
Academic	94	38.30	36.17	3.19	42.55	52.13
Agent history QA	20	35.00	10.00	0.00	35.00	50.00
Code repo QA	50	32.00	40.00	4.00	14.00	48.00
Detective	22	9.09	13.64	0.00	4.55	18.18
Dialogue history QA	19	31.58	36.84	15.79	31.58	42.11
Event ordering	20	40.00	5.00	0.00	40.00	65.00
Financial	37	32.43	37.84	0.00	37.84	45.95
Governmental	41	24.39	26.83	0.00	17.07	34.15
Knowledge graph reasoning	15	20.00	40.00	0.00	66.67	66.67
Legal	33	33.33	33.33	3.03	39.39	36.36
Literary	30	20.00	26.67	10.00	40.00	50.00
Many-shot learning	21	23.81	19.05	33.33	28.57	57.14
Multi-news	23	17.39	17.39	0.00	21.74	26.09
New language translation	20	35.00	5.00	20.00	30.00	35.00
Table QA	18	33.33	11.11	5.56	5.56	22.22
User guide QA	40	27.50	25.00	10.00	40.00	42.50

G.5 MAIN: LONGBENCH-V2 BY DIFFICULTY

Table 19: Main LongBench-v2 by difficulty: official score (%). Accuracy by the indicated category.

Difficulty	n	BM25	LLMx	RA-P	Direct-HT	TopoMem
easy	192	29.69	29.17	6.77	36.46	45.83
hard	311	29.90	26.37	4.82	28.62	43.09

G.6 MAIN: LONGBENCH-V2 BY LENGTH

Table 20: Main LongBench-v2 by length: official score (%). Accuracy by the indicated category.

Length	n	BM25	LLMx	RA-P	Direct-HT	TopoMem
long	108	31.48	26.85	8.33	10.19	34.26
medium	215	28.84	23.72	7.91	30.70	46.98
short	180	30.00	32.22	1.11	45.56	46.67

G.7 MAIN: INFINITEBENCH BY TASK

Table 21: Main InfiniteBench by task: official score (%). Task-specific official metrics retain their original definitions; scores are not uniformly exact-match accuracy. The task-balanced mean weights the 12 task means equally; the micro mean weights examples equally.

Task	n	BM25	LLMx	RA-P	Direct-HT	TopoMem
code_debug	394	32.99	–	28.93	52.54	34.26
code_run	400	2.75	–	0.50	2.00	100.00
kv_retrieval	500	99.40	–	77.00	85.80	99.80
longbook_choice_eng	229	46.72	–	62.01	59.83	74.67
longbook_qa_chn	189	22.10	–	25.80	25.72	45.63
longbook_qa_eng	351	23.66	–	21.39	34.12	28.29
longbook_sum_eng	103	20.42	–	15.86	22.17	28.95
longdialogue_qa_eng	200	32.50	–	13.00	53.50	29.50
math_calc	50	0.00	–	0.00	0.00	100.00
math_find	350	38.29	–	27.71	39.71	30.00
number_string	590	100.00	–	100.00	100.00	100.00
passkey	590	100.00	–	100.00	100.00	100.00

G.8 MAIN: RULER BY TASK AND LENGTH

Table 22: Main RULER by task and length: official score (%). All 13 tasks and all three evaluated scales are shown.

Task	Length	n	BM25	LLMx	RA-P	Direct-HT	TopoMem
cwe	8192	100	52.40	–	98.00	99.90	100.00
cwe	32768	100	97.00	–	99.00	99.90	100.00
cwe	131072	100	1.30	–	98.40	99.10	100.00
fwe	8192	100	100.00	–	95.67	99.67	100.00
fwe	32768	100	100.00	–	99.00	100.00	100.00
fwe	131072	100	99.33	–	100.00	99.33	100.00
niah_multikey_1	8192	100	100.00	–	100.00	100.00	100.00
niah_multikey_1	32768	100	98.00	–	100.00	100.00	100.00
niah_multikey_1	131072	100	97.00	–	99.00	99.00	100.00
niah_multikey_2	8192	100	100.00	–	46.00	100.00	100.00
niah_multikey_2	32768	100	100.00	–	31.00	100.00	100.00
niah_multikey_2	131072	100	100.00	–	29.00	94.00	100.00
niah_multikey_3	8192	100	100.00	–	53.00	100.00	100.00
niah_multikey_3	32768	100	100.00	–	18.00	100.00	100.00
niah_multikey_3	131072	100	100.00	–	5.00	89.00	100.00
niah_multiquery	8192	100	100.00	–	100.00	99.75	99.75
niah_multiquery	32768	100	100.00	–	100.00	100.00	100.00
niah_multiquery	131072	100	100.00	–	100.00	100.00	100.00
niah_multivalue	8192	100	100.00	–	100.00	100.00	100.00
niah_multivalue	32768	100	100.00	–	100.00	100.00	100.00
niah_multivalue	131072	100	100.00	–	99.75	99.75	100.00
niah_single_1	8192	100	100.00	–	100.00	100.00	99.00
niah_single_1	32768	100	100.00	–	100.00	100.00	99.00
niah_single_1	131072	100	100.00	–	100.00	100.00	99.00
niah_single_2	8192	100	100.00	–	100.00	100.00	100.00
niah_single_2	32768	100	100.00	–	100.00	100.00	98.00
niah_single_2	131072	100	100.00	–	100.00	100.00	100.00
niah_single_3	8192	100	100.00	–	100.00	100.00	100.00
niah_single_3	32768	100	100.00	–	100.00	100.00	100.00
niah_single_3	131072	100	100.00	–	100.00	100.00	99.00
qa_1	8192	100	90.00	–	89.00	92.00	87.00
qa_1	32768	100	84.00	–	66.00	93.00	82.00
qa_1	131072	100	68.00	–	55.00	92.00	78.00
qa_2	8192	100	69.00	–	56.00	65.00	59.00
qa_2	32768	100	60.00	–	56.00	71.00	60.00
qa_2	131072	100	55.00	–	47.00	58.00	47.00
vt	8192	100	0.00	–	94.60	100.00	100.00
vt	32768	100	80.80	–	96.20	99.60	100.00
vt	131072	100	18.00	–	97.20	100.00	100.00

G.9 MAIN: BABILONG BY TASK AND LENGTH

Averaged over four lengths, TopoMem gains 42.75, 77.75, 30.50, and 28.50 points over Direct-HT on QA1–QA4, respectively, and trails by 13.75 points on QA5. QA3 and QA5 account for 83.4% of TopoMem's remaining gap to a 100% score despite comprising 40% of the evaluated examples.

Table 23: Main BABILong by task and length: official score (%). Task identifiers follow the benchmark. Preliminary cells have one example; main cells have 100 examples. BM25 and RA-P use Ascend inference; LLMx and TopoMem use H200 inference.

Task	Length	n	BM25	LLMx	RA-P	Direct-HT	TopoMem
qa1	16k	100	46.00	71.00	12.00	56.00	97.00
qa1	32k	100	39.00	61.00	12.00	54.00	97.00
qa1	64k	100	31.00	55.00	16.00	56.00	97.00
qa1	128k	100	28.00	47.00	17.00	51.00	97.00
qa2	16k	100	0.00	5.00	0.00	14.00	94.00
qa2	32k	100	1.00	4.00	2.00	17.00	94.00
qa2	64k	100	7.00	2.00	0.00	19.00	94.00
qa2	128k	100	8.00	0.00	1.00	15.00	94.00
qa3	16k	100	11.00	8.00	3.00	29.00	56.00
qa3	32k	100	10.00	5.00	4.00	30.00	56.00
qa3	64k	100	8.00	1.00	5.00	26.00	57.00
qa3	128k	100	8.00	1.00	3.00	18.00	56.00
qa4	16k	100	57.00	86.00	8.00	64.00	95.00
qa4	32k	100	54.00	92.00	6.00	59.00	93.00
qa4	64k	100	55.00	96.00	9.00	64.00	91.00
qa4	128k	100	51.00	86.00	6.00	70.00	92.00
qa5	16k	100	65.00	75.00	29.00	77.00	67.00
qa5	32k	100	60.00	80.00	19.00	77.00	59.00
qa5	64k	100	45.00	78.00	28.00	73.00	61.00
qa5	128k	100	51.00	79.00	18.00	76.00	61.00

G.10 BM25 RETRIEVAL SENSITIVITY ON BABILONG

The main BM25 baseline retrieves eight chunks. Diagnostic variants use $k \in \{1, 2, 4\}$ on the same 2,000 BABILong examples, with unchanged chunks, ranker, reader, decoding, and official scoring. All use Ascend TP1. Repeating the k=4 and k=8 evaluations on a second host produced identical raw predictions for all 2,000 examples. On that host, k=4 gains 3.10 points over k=8 (task–length-stratified paired 95% CI [1.85, 4.35]; 116 wins, 54 losses, 1,830 ties).

Table 24: BM25 retrieval-depth diagnostic on BABILong (%). Parentheses give the task–length-stratified paired difference from the main k=8 point and its 95% bootstrap interval.

k	Overall	QA1	QA2	QA3	QA4	QA5
1	30.20 (-1.55; [-3.50, 0.35])	43.25	0.00	3.00	55.50	49.25
2	32.80 (+1.05; [-0.60, 2.70])	42.50	0.25	5.75	58.00	57.50
4	**34.85** (+3.10; [1.90, 4.35])	41.50	2.00	9.00	59.00	62.75
8 (main)	31.75	36.00	4.00	9.25	54.25	55.25

At a nominal 32k retrieval budget, 4k×8 and 16k×2 differ from 8k×4 by -0.50 points (task–length-stratified 95% CI [-1.80, 0.80]) and -0.35 ([-1.70, 1.00]). With 8k chunks, replacing independent top-4 by top-2 anchors plus one-hop neighbors scores 33.75% (-1.10; [-2.25, 0.05]); adding first/last chunks instead scores 33.55% (-1.30; [-2.45, -0.20]; 52 wins, 78 losses, 1,870 ties). None of the tested chunk-size, neighbor, or boundary variants closes the performance gap. The nominal budget control remains approximate.

Table 25: The same diagnostic by context length, with observed reader input and median enclosing time. Token totals cover all 2,000 examples.

k	16k	32k	64k	128k	Prompt tok. (M)	Median s
1	34.20	32.40	30.60	23.60	13.78	1.293
2	35.40	33.20	31.80	30.80	27.52	2.006
4	35.80	34.40	35.60	33.60	49.07	3.389
8 (main)	35.80	32.80	29.20	29.20	78.61	4.786

Table 26: Fixed nominal 32k retrieval-budget chunk-granularity diagnostic. Prompt-token totals include unchanged scaffolding over all 2,000 examples.

Chunk$\times k$	Overall	16k	32k	64k	128k	Prompt tok. (M)	Median s
4k$\times$8	34.35	35.60	35.00	34.40	32.40	49.08	3.375
8k$\times$4	**34.85**	35.80	34.40	35.60	33.60	49.07	3.389
8k neighbors	33.75	35.80	35.40	33.60	30.20	48.64	3.370
8k boundary	33.55	35.80	34.40	33.60	30.40	46.34	3.141
16k$\times$2	34.50	36.00	34.80	33.80	33.40	50.15	3.555

G.11 Context-window fit and direct-inference cost

Direct-HT produced nonempty predictions for all 10,574 main-evaluation examples. Table 27 separates full-context-fit rows from rows requiring the deterministic head–tail rule. LongBench-v2 and InfiniteBench contain the 392 over-window examples.

Table 27: Direct-HT coverage and official scores (%). InfiniteBench reports its task-balanced mean with the micro mean in parentheses. "Overflow" denotes deterministic symmetric head–tail truncation, not a dropped row.

Suite	n	Full-context fit	Overflow	Score
NIAH	225	225	0	100.00
LongBench-v2	503	403	100	31.61
InfiniteBench	3,946	3,654	292	47.95 (60.78)
RULER	3,900	3,900	0	96.15
BABILong	2,000	2,000	0	47.25

Within LongBench-v2, Direct-HT scores 36.23% on the 403 full-context-fit examples and 13.00% on the 100 over-window examples. TopoMem scores 45.91% and 37.00% on the same respective IDs, for paired gains of 9.68 points (95% bootstrap CI [4.22, 15.14]) and 24.00 points ([14.00, 34.00]). The former stratum shows that TopoMem's LongBench-v2 gain is not solely an artifact of Direct-HT's truncation policy; the latter quantifies the additional gap when the direct prompt exceeds 262K. On InfiniteBench, Direct-HT scores 63.56% on 3,654 fit examples and 25.90% on 292 overflow examples; TopoMem scores 73.70% and 41.53% on the same IDs, for paired micro-score gains of 10.14 points (95% CI [8.55, 11.73]) and 15.63 points ([9.90, 21.40]).

Table 28: Observed Direct-HT computation. Each example makes one model request; time is median enclosing time per example. The pooled five-suite median is not available from the retained summary records.

Suite	Requests	Prompt tok.	Completion tok.	Median time (s)
NIAH	225	14,765,957	5,028	8.601
LongBench-v2	503	62,203,472	6,208	14.094
InfiniteBench	3,946	546,989,597	1,208,517	17.909
RULER	3,900	219,606,651	82,358	3.884
BABILong	2,000	116,390,823	10,823	5.120
Total	10,574	959,956,500	1,312,934	–

G.12 Execution-route usage and paired ablations

Across the 10,574 main-evaluation examples, TopoMem uses 13 execution routes. Table 29 groups them by memory family, and Table 30 gives each route's frequency and main computation.

Table 29: Execution-family coverage over all main-evaluation predictions. "Typed" includes deterministic state and typed reducers; "structured" denotes model-backed structured readers.

Suite	Typed	Structured	Neural retrieval	Total
NIAH	0	0	225	225
LongBench-v2	0	503	0	503
InfiniteBench	722	336	2,888	3,946
RULER	1,205	0	2,695	3,900
BABILong	1,452	0	548	2,000
Total	3,379	839	6,356	10,574

Table 30: The 13 execution routes observed among 10,574 main-evaluation predictions. n counts routed examples; descriptions give the main memory or action of each route.

Execution route	n	Main memory or action
`STATE_EXECUTE`	1,461	Ordered event state; event-log reader if needed
`COLLECT`	305	Deduplicated relation-value set
`TRACE`	300	Transitive closure over assignment edges
`COUNT_TOPK`	600	Global item counts and top-k
`REDUCE`	263	Sum, mean, min, max, or count of extracted values
`SLICE_EXECUTE`	50	Intermediate values in an arithmetic stream
`JOIN`	400	Evaluation of supported function dependencies
`ORDER`	4	Sorted dated or indexed evidence for a reader
`COMPRESS`	103	Rolling comprehensive semantic memory
`VERIFY_SELECT`	732	Option-wise evidence ledger and adjudication
`SPARSE_LOCATE`	5,921	Eight ranked evidence windows for a reader
`MULTI_LOCATE`	41	Twelve ranked windows for multi-evidence queries
`DEBUG_LOCATE`	394	Fault-cue/identifier-option sparse reader

Typed execution is concentrated in synthetic RULER/BABILong (2,657 rows) but is not confined there: InfiniteBench contributes 722 JOIN, REDUCE, SLICE_EXECUTE, or STATE_EXECUTE rows. Conversely, NIAH uses only sparse lookup and LongBench-v2 only closed-choice verification. ORDER (4) and MULTI_LOCATE (41) are rare, so evidence is much thinner for these routes than for the commonly used operators.

The 620-example paired ablation subset uses six routes. Typed operators cover 162 examples at 98.77%, neural lookup covers 358 at 90.78%, and VERIFYSELECT covers 100 LongBench-v2 examples at 48.00%. The paired analysis below compares end-to-end variants on these same 620 examples.

Across the same 620 examples, full TopoMem scores 85.97% versus 71.83% without topology, a +14.14-point paired difference (task-stratified 95% bootstrap CI [12.32, 15.93]; 106 wins, 16 losses, 498 ties). Table 31 groups examples by their full-method route. Each row describes heterogeneity in the end-to-end difference rather than an isolated operator effect.

Table 31: Topology-removal difference by full-method route (620 paired examples). Intervals are task-stratified paired 95% bootstrap CIs; W/L/T counts compare per-example official scores.

Full-side route	n	Full	No topology	Δ [95% CI]	W/L/T
COUNT_TOPK	80	100.00	50.71	+49.29 [46.88, 51.00]	41/0/39
COLLECT	42	95.24	99.40	−4.17 [−4.76, −2.98]	1/2/39
TRACE	40	100.00	0.00	+100.00 [100.00, 100.00]	40/0/0
SPARSE_LOCATE	353	91.22	91.22	0.00 [0.00, 0.00]	0/0/353
MULTI_LOCATE	5	60.00	60.00	0.00 [0.00, 0.00]	0/0/5
VERIFY_SELECT	100	48.00	38.00	+10.00 [0.00, 20.00]	24/14/62

The `COLLECT` row contains 40 `niah_multivalue` examples and one each from `qa_1` and `qa_2`; the two negative singleton task strata remain fixed under the task-stratified bootstrap. A post-hoc unstratified paired bootstrap over the same 42 IDs (20,000 replicates; seed 20260926) gives the same -4.17-point estimate but a 95% interval of $[-11.90, 1.19]$, rather than $[-4.76, -2.98]$. Both intervals are reported because their different resampling assumptions change whether the interval includes zero.

The 106 favorable transitions concentrate in typed operators (82: 41 COUNT_TOPK, 40 TRACE, and one COLLECT) and VERIFY_SELECT (24); all 16 reverse transitions are COLLECT (2) or VERIFY_SELECT (14). Neural-locate predictions are identical under this intervention because no-topology uses the same fixed sparse retriever there. Consequently the observed gain is associated with selecting executable counting/tracing operators rather than a generic reader improvement. A paired three-way comparison sharpens this interpretation: removing verification leaves all 520 non-VERIFY_SELECT rows exactly unchanged, including the 82 RULER gains over no-topology. On the 100 VERIFY_SELECT rows, full scores 48.00% versus 53.00% without verification (-5.00 points; task-stratified 95% CI $[-15.00, 5.00]$; 13 wins, 18 losses, 69 ties). The point estimate favors no verification, but its interval spans zero and the variant also changes evidence budgeting.

H Additional baseline and computation details

H.1 Preliminary evaluation of additional baselines

Table 32 compares five methods on the same 330 preliminary examples. LongLLMLingua and CoA have no results on the larger main benchmark sets. CoA was run on CUDA, and the preliminary ReadAgent-P row combines Ascend and H200 inference; common examples and backbone do not establish backend equivalence. Published results under other protocols appear separately in Appendix H.3.

Preliminary InfiniteBench uses a micro mean, unlike the task-balanced main number in Table 1. Preliminary BABILong contains only 20 QA1–QA10 examples at 32k/128k; the main suite contains 2,000 QA1–QA5 examples at four lengths.

Across the five methods in Table 32, TopoMem exceeds LLM×MapReduce-v1 on four families and ties on NIAH, and it exceeds LongLLMLingua on all five families. For the paired TopoMem–LLM×MapReduce-v1 comparison, the unweighted mean of the five family differences is 22.14 points (stratified paired 95% bootstrap CI [15.38, 28.86]); the LongBench-v2 difference remains unresolved (13.33 points; CI $[-10.00, 36.67]$). LongLLMLingua's row scores every observed empty output as zero rather than dropping its nine method failures or two overflows. TopoMem also exceeds CoA on all five preliminary families; CoA was run on CUDA. The unweighted mean of the five TopoMem–CoA family differences is 46.36 points (stratified paired 95% bootstrap CI [39.07, 53.31]), while their LongBench-v2 difference remains unresolved (6.67 points; CI $[-16.67$, 30.00]). These preliminary results do not establish a main-benchmark advantage. ReadAgent-P retains its one method failure as a zero-scored example.

Table 32: Preliminary Qwen3.5-9B scores on 330 shared examples (%). All columns are per-example micro means. The ReadAgent-P row combines Ascend and H200 inference; CoA uses CUDA. Shared IDs and backbone do not establish inference-backend equivalence.

Method	NIAH	LongBench-v2	InfiniteBench	RULER	BABILong
LLM×MapReduce-v1	100.00	36.67	47.28	78.05	45.00
LongLLMLingua	72.00	20.00	39.78	47.55	30.00
CoA	44.00	43.33	33.09	55.49	10.00
ReadAgent-P	56.00	13.33	37.62	80.53	10.00
TopoMem	**100.00**	**50.00**	**71.79**	**95.90**	**100.00**

H.2 ReadAgent-P main benchmark comparisons

ReadAgent-P predictions were generated on Ascend and TopoMem predictions on H200. The following comparisons use the same evaluation examples and official metrics, but the inference environments and method budgets differ. On 225 NIAH examples, ReadAgent-P answers 49 correctly versus TopoMem's 225, a 78.22-point paired difference (95% CI [72.44, 83.56]). On 503 LongBench-v2 examples, the counts are 28 and 222, a 38.57-point difference ([33.80, 43.34]). Nine ReadAgent-P context-window failures score zero. Among its remaining 494 LongBench-v2 predictions, 416 lack an extractable A–D option, 50 extractable options are wrong, and 28 are correct. Its 5.57% score therefore reflects both output-format compatibility and answer correctness; it cannot isolate reasoning quality.

The BABILong and RULER paired intervals use 20,000 bootstrap replicates over their respective example sets.

On 2,000 BABILong examples, ReadAgent-P scores 9.90% versus TopoMem's 80.40%, a 70.50-point paired difference (95% CI [68.30, 72.65]). On 3,900 RULER examples, the paired difference is 12.28 points ([11.10, 13.46]); TopoMem wins 605 cases, ReadAgent-P wins 65, and 3,230 tie. On all 3,946 InfiniteBench examples, the task-balanced difference is 24.91 points ([23.52, 26.30]) and the micro difference is 18.45 points ([16.90, 20.00]); TopoMem wins 1,134 examples, ReadAgent-P wins 374, and 2,438 tie. Five ReadAgent-P inference errors remain zero-scored. These are same-example score differences, not inference-backend-equivalent comparisons.

H.3 Published results under different evaluation protocols

These published numbers provide method context only; they are not predictions on our main or preliminary evaluation sets. Zhou et al. (2025), Table 1, report nine-task InfiniteBench averages with Llama3-70B-Instruct, temperature 0.7, Code.Run and Math.Calc excluded, and one task manually assessed. Their CoA and LongAgent rows are their reproductions rather than original-author results. Li et al. (2025), Table 4, report LongBench-v2 micro averages with Qwen2-7B-Instruct at $4\times$ compression; their LongLLMLingua, SnapKV, and Activa-

tion Beacon rows are third-party reproductions. Exact item/scorer parity with our 503-question manifest is not established.

Table 33: Published comparison-method scores (%). The two source protocols and backbones differ from our Qwen3.5-9B evaluation; no TopoMem score is inserted and no cross-panel ranking or paired comparison is implied.

InfiniteBench: Llama3-70B-Instruct, nine tasks (Zhou et al. (2025), Table 1)	
Method	Reported score
CoA	15.97
LongAgent	53.81
LLM×MapReduce	68.66

LongBench-v2: Qwen2-7B-Instruct, 4× compression (Li et al. (2025), Table 4)	
Method	Reported micro score
LongLLMLingua	18.10
SnapKV	27.90
Activation Beacon	28.30
AdmTree	31.90

H.4 Observed computation and accounting boundary

Every model request in the ablation runs was recorded. Table 34 shows lower model-call counts for the fixed sparse variant alongside lower RULER accuracy. The no-verification variant records 100 fewer calls than the fixed-budget variant, but the two also differ in budgeting. These durations do not measure batch wall-clock time, energy, or monetary cost.

Table 34: Complete observed computation for each 620-example ablation run. Tokens are totals; time is the median recorded duration per example in seconds.

Variant	Calls	Prompt tok.	Completion tok.	Median time (s)
No topology	620	3,932,799	13,989	1.745
No verification	2,996	7,841,225	96,133	9.583
Fixed evidence budget	3,096	12,783,925	187,978	10.136

The main-run instrumentation did not record semantic-compilation calls. Its 620-example totals (1,436 requests, 12.33M prompt tokens, and 129.7K completion tokens) are therefore lower bounds, not a full-method cost comparison with the ablations. Official accuracy uses all saved predictions.

On 225 NIAH and 503 LongBench-v2 examples, LLM×MapReduce-v1 records 54,359 requests (51,584 map, 2,047 collapse, 728 reduce), 187,465,361 prompt tokens, and 20,019,326 completion tokens. Comparable end-to-end wall-clock measurements are unavailable for this run. Direct-HT records 10,574 requests, 959,956,500 prompt and 1,312,934 completion tokens. LLM×MapReduce-v1's BABILong counters cover 1,999/2,000 rows: 39,368 requests, 132,881,106 prompt and 9,857,081 completion tokens, and 162,344.5 summed API-latency seconds. Its empty-prediction row lacks counters. ReadAgent-P's 2,000 BABILong records preserve 350,856 requests, 485,275,344 prompt and 39,532,989 completion tokens, excluding interrupted calls. RULER's 3,900 evaluated examples add 508,137 requests, 933,630,030 prompt and 46,916,783 completion tokens, and 1,701,980.9 summed API-latency seconds. ReadAgent-P InfiniteBench has counters for 3,941/3,946 evaluated examples; its five retained errors have no counters. Observable lower bounds are 1,760,776 requests, 3,044,648,906 prompt tokens, 156,563,140 completion tokens, and 5,626,699.9 summed API-latency seconds. These are lower bounds excluding interrupted calls, not wall time or device-hours; cross-hardware efficiency is not established.