

Memory Architectures Beyond Attention:

Disambiguating Four Memory Concepts and the Long-Context Reasoning Frontier

Authors

Lead Researcher: Pranay M Mahendrakar

Mail: pranaymahendrakar2001@gmail.com

Contact: +91 6361723454

Company: SONYTECH

Date: April 2026

Abstract

State-space models did not merely "open a door" beyond attention; they walked through it. Mamba, Mamba-2, Samba, Jamba, Granite 4, Hymba, and a growing family of hybrid attention-SSM architectures are now in production language models, demonstrating that linear-time alternatives to attention can match transformers on language modelling at competitive scales. The simultaneous expansion of pure-attention context windows — Gemini 1.5 and successors handling up to 10 million tokens with near-perfect needle-in-haystack recall — has changed the empirical landscape that motivated SSM research in the first place. The popular framing of "memory architectures beyond attention" has not kept up with this. This paper makes three claims. First, the word "memory" in the long-context discussion conflates four distinct concepts — architectural state, context window, external retrieval, and persistent agent memory — each with different scaling properties and different research questions. Second, the empirical picture is more nuanced than either the SSM-replaces-attention or the attention-is-enough framings: SSMs win on very long passive recall and inference efficiency, transformers win on complex reasoning, hybrids win in deployment, and the choice between them is task-dependent. Third, the genuine open frontier is reasoning at long context — not retrieval, which is largely solved — and benchmarks like MathHay (51% accuracy at 128K tokens for Gemini-1.5-Pro) make the gap quantitatively visible. We propose a research agenda focused on architecture-task fit, hierarchical multi-scale memory, and reasoning-aware long-context evaluation.

Keywords

State-space models, Mamba, hybrid architectures, long-context language models, attention, memory, RAG, agentic memory, long-context reasoning

1. Introduction

The standard narrative is familiar. Transformers dominated language modelling for half a decade, attention's quadratic complexity in sequence length became increasingly painful at very long contexts, and state-space models (SSMs) — Mamba (Gu & Dao, 2023), Mamba-2 (Dao & Gu, 2024), and the broader S4 family (Gu et al., 2022) — emerged as a linear-time alternative that compressed prior context into a fixed-size hidden state. The story is often told as a transition still in progress: SSMs as a promising direction, attention as the entrenched incumbent, the field waiting to see which architecture wins.

This narrative has not kept pace with the actual state of the art. As of 2026, hybrid attention-SSM architectures are in production language models. Jamba 1.5 (AI21 Labs), Samba (Ren et al., 2024), IBM's Granite 4, NVIDIA's Hymba, and the broader hybrid family combine attention layers with Mamba

or related SSM blocks at fixed ratios or with learned routing between them. Pure transformers, meanwhile, have not stood still: Gemini 1.5 Pro (Reid et al., 2024) handles up to ten million tokens of context with near-perfect needle-in-haystack recall, demonstrating that scaling tricks on attention can reach context lengths SSMs were initially proposed to address. The competition between architectures has produced winners in different niches rather than a clear successor to attention.

At the same time, the popular framing of "memory beyond attention" obscures a real conceptual confusion. The term "memory" in the long-context literature is used for at least four distinct things: the architectural state of an SSM or the KV cache of a transformer, the inference-time context window, external retrieval-augmented generation (RAG) over a vector database, and the persistent memory of an agent across sessions. These have different scaling characteristics, different research bottlenecks, and different

relevant interventions. Conflating them produces incoherent research agendas and misleading benchmark comparisons.

This paper argues that the genuine open problem is not memory architectures per se but reasoning at long context. Long-context retrieval — finding a fact in a million tokens — is largely solved by current systems. Long-context reasoning — performing multi-step inference over information distributed across long contexts — is not. MathHay (Wang et al., 2024) demonstrates this concretely: even Gemini-1.5-Pro-002 achieves only 51.26% accuracy on long-context mathematical reasoning at 128K tokens, well below its near-perfect needle recall at substantially longer contexts. The gap between retrieval-at-long-context (solved) and reasoning-at-long-context (open) is the most informative way to characterise where the field actually stands.

Section 2 surveys where each architectural family stands. Section 3 develops the four-memory-concepts disambiguation. Section 4 describes the empirical picture across families. Section 5 identifies three productive frontiers. Section 6 proposes a research agenda. The contribution is structural: a clearer framing of an area where the popular narrative has not kept pace with the empirical reality.

2. Where Each Architectural Family Stands

2.1 Attention: Painful at Scale, Still Dominant

The transformer's quadratic complexity in sequence length is real but has been pushed back further than the SSM literature sometimes acknowledges. FlashAttention (Dao et al., 2022) reduced the constant factor substantially. Sparse and sliding-window variants (Longformer, BigBird) handle moderate context. KV-cache compression and quantisation extend deployment context limits. Most importantly, frontier model providers have demonstrated that attention can scale to context lengths far beyond what the quadratic complexity alone would suggest is feasible: Gemini 1.5 reports near-perfect needle recall up to 10M tokens, Claude and GPT families have moved from 4K to 128K to 200K-plus context windows over a few years.

These extensions come with caveats — performance on retrieval at the extreme tail of context, latency, memory cost, and the well-documented gap between retrieval and reasoning. But the framing that attention has hit a wall at long context is not consistent with current capability.

2.2 State-Space Models: Linear-Time Alternative

S4 (Gu et al., 2022) established that structured state-space models could be trained stably and efficiently at deep-

learning scale. Mamba (Gu & Dao, 2023) added input-dependent selectivity, addressing the SSM literature's longstanding limitation that fixed-state recurrence cannot adaptively decide what to remember. Mamba-2 (Dao & Gu, 2024) generalised this further, exposing a structural relationship between SSMs and attention that enabled larger state sizes and more efficient training. Pure SSM language models at scales up to several billion parameters now exist and are competitive with transformer baselines on standard language modelling benchmarks.

The honest weaknesses are also documented. SSMs, due to their recurrent nature with fixed-dimensional state, struggle with precise memory recall — the canonical evidence is that pure Mamba models underperform transformers on associative-recall tasks (Arora et al., 2023; Waleffe et al., 2024). The fixed state must compress all prior context, which works well for distributional information and language patterns but degrades for tasks requiring exact retrieval of specific tokens or facts.

2.3 Hybrid Architectures: Production Reality

Recognising the complementary strengths of attention (precise recall, complex reasoning) and SSMs (linear-time long-context efficiency), hybrid architectures interleave layers of each type. Jamba (Lieber et al., 2024) was the first widely deployed example, combining transformer and Mamba layers with mixture-of-experts. Samba (Ren et al., 2024) developed the design space further with explicit attention to length extrapolation. IBM's Granite 4 (2024) and NVIDIA's Hymba represent the design's adoption by large industrial labs. TransMamba (2025) and MaBERT (2026) explore variations including dynamic switching between attention and SSM, sequence-level hybridisation, and bidirectional encoder applications.

Empirical results suggest that 1:6 to 1:8 attention-to-SSM layer ratios capture most of the benefit of attention while preserving most of the efficiency of SSMs. This is the current production sweet spot. Whether the optimal ratio depends on task, scale, or context length is a live research question.

2.4 Linear Attention and Related Variants

A parallel line of work has developed linear-time alternatives to standard attention without departing from the attention framework. Linear Transformers (Katharopoulos et al., 2020), Performers, RWKV (Peng et al., 2023), and Retentive Networks (Sun et al., 2023) all reformulate attention to achieve linear-time inference, typically through kernel approximations or recurrent reformulations. Gated Linear Attention (Yang et al., 2024) and DeltaNet are recent additions. These approaches occupy a design space adjacent

to SSMs and produce models with similar inference efficiency, sometimes with different training dynamics.

3. Four Distinct Concepts Called "Memory"

The discussion of memory in long-context AI conflates four distinct things. Each has different scaling characteristics, different research bottlenecks, and different relevant interventions.

3.1 Architectural State

This is the internal state the model uses during a single forward pass: the KV cache for attention, the hidden state for SSMs, recurrent state for RNN-like architectures. Architectural state is what determines per-token inference cost and what bounds the model's working memory at any moment. The transition from quadratic to linear architectural-state cost is what SSMs and linear attention promise. Research questions here are about model architecture: state dimensionality, update rules, gating mechanisms, hardware-friendly implementations.

3.2 Context Window

This is the maximum number of tokens the model can consume in a single inference call. Context window size is determined partly by architecture (transformers need long context to be tractable; SSMs handle it natively) and partly by training (positional encoding choices, length extrapolation methods). Gemini 1.5's 10M-token context, Claude's 200K, GPT's 128K are context-window numbers. Research questions here are about positional encodings, length extrapolation, training procedures for long context, and inference engineering.

3.3 External Retrieval

Retrieval-augmented generation (RAG) systems use external vector databases to fetch relevant content into the context window at inference time. The "memory" here is not in the model at all; it is in an external store the model queries. RAG addresses problems of unbounded knowledge and freshness without architectural changes to the model. Research questions are about retrieval quality, chunking strategies, embedding models, and the interaction between retrieval and generation. RAG is largely an orthogonal axis to architectural memory choices.

3.4 Persistent Agent Memory

Recent agent frameworks (Mem0, Letta, MemGPT-style architectures) maintain memory across sessions: facts about the user, prior conversation summaries, learned preferences. This memory is typically implemented as structured external storage — sometimes a vector database, sometimes a graph, sometimes a structured key-value store

— that the agent reads from and writes to during operation. Research questions are about memory selection, summarisation, conflict resolution, privacy, and the interface between persistent memory and in-context working memory.

These four are not interchangeable. Adding RAG does not solve architectural-state limits. Extending context window does not give the model persistent memory. Improving SSM hidden-state design does not address what an agent should remember between sessions. Conflating them in research papers leads to comparisons that are not measuring what they appear to be measuring.

4. The Empirical Picture: What Works Where

4.1 SSMs Win at Very Long Passive Recall

On tasks where the model must process very long sequences with relatively simple downstream demands — long-context language modelling perplexity, long-form summarisation, simple retrieval — SSMs and hybrids match or exceed transformer baselines while using a fraction of the compute and memory. The selectivity in Mamba addresses the early SSM weakness on input-dependent retention. For these workloads, the architecture-driven efficiency advantage is real and material.

4.2 Transformers Still Win on Complex Reasoning

On tasks requiring precise recall combined with multi-step reasoning — mathematical reasoning over long contexts, code understanding across large repositories, in-context learning from long examples — pure transformers and hybrids with substantial attention proportions outperform pure SSMs. The mechanism is plausible: precise associative recall benefits from the unrestricted access pattern of attention; reasoning over many specific items benefits from explicit pairwise interactions. SSMs' fixed-state compression hurts here in ways that are not fully addressed by selectivity alone.

4.3 Hybrids Win in Production

The deployment evidence is consistent: hybrid architectures dominate the production frontier. Jamba 1.5 in commercial use, Samba in benchmarks, Granite 4 from IBM, Hymba from NVIDIA — these are not research curiosities but shipping models. The reason is straightforward: hybrids capture most of the SSM efficiency advantage at long context while preserving enough attention to handle the reasoning workloads where pure SSMs lag. The 1:6 to 1:8 attention-to-SSM ratio is currently a useful default; whether it remains so as workloads evolve is open.

4.4 The Reasoning-at-Long-Context Gap

This is the empirical finding that most informs research priorities. Long-context retrieval is largely solved at the frontier: Gemini 1.5 hits near-perfect needle-in-haystack at 10M tokens. Long-context reasoning is not. MathHay (Wang et al., 2024) reports 51.26% accuracy for Gemini-1.5-Pro-002 on mathematical reasoning at 128K tokens. Recent extended needle-in-haystack benchmarks (Liu et al., 2026) find that performance degrades substantially when relevant evidence is diluted or widely dispersed and the task requires logical inference rather than literal extraction. The Needle Threading benchmark (Roberts et al., 2024) requires following chains of linked information across long contexts and exposes performance gaps not visible in single-needle retrieval.

The retrieval-versus-reasoning gap is the most useful framing of where the field's open problem actually lives. Memory-architecture research that improves retrieval performance at very long contexts addresses a problem largely solved by current systems. Memory-architecture research that improves multi-step reasoning at long context addresses a problem that is genuinely open.

5. Three Productive Frontiers

5.1 Architecture Co-Design for Reasoning + Recall

Current hybrid architectures choose attention-to-SSM ratios at design time and leave them fixed across all inputs. A reasoning-heavy task at moderate context gets the same architectural budget as a retrieval-heavy task at extreme context. Dynamic routing — letting the model select which layers to use depending on the input or per-token requirements — is a research direction with clear theoretical motivation and limited empirical exploration. TransMamba (2025) is one early example; the design space for adaptive hybrid routing is barely sketched.

5.2 Test-Time Training and Inference-Time Memory Updates

Test-Time Training (TTT) layers (Sun et al., 2024) propose hidden states that are themselves small learnable models updated at inference time. This blurs the distinction between architectural state and learned weights and offers a route toward memory that adapts to context patterns within a single sequence. Whether TTT-class methods scale and whether their advantages survive at larger model sizes are open questions, but the conceptual move — letting memory itself be learned at inference rather than fixed at training — is likely to produce useful architectures regardless of which specific instantiation wins.

5.3 Hierarchical and Multi-Scale Memory

Human cognition arguably uses memory at multiple time scales — working memory of seconds, episodic memory of hours, semantic memory of lifetime. Current language model architectures collapse these into one mechanism. Hierarchical architectures that maintain memory at distinct scales — fast-updating local working memory, slower-updating episodic summaries, persistent semantic state — are an under-explored direction. Some agent-memory systems (MemGPT-class) approximate this externally; doing it natively in architecture is largely open.

6. A Research Agenda

6.1 Methodological Commitments

Three commitments would help. First, papers comparing memory architectures should be explicit about which of the four memory concepts they address. A paper comparing Mamba to a transformer on architectural-state efficiency is doing different work from a paper comparing RAG to long-context inference, and should not be reported as if they were the same comparison. Second, long-context evaluations should distinguish retrieval from reasoning. The needle-in-haystack family is a useful retrieval benchmark; treating it as a measure of long-context capability writ large is misleading given how much better current systems are at retrieval than at reasoning. Third, hybrid architecture comparisons should report performance stratified by task type — the attention-to-SSM ratio that wins on perplexity may not be the ratio that wins on mathematical reasoning, and reporting only one understates the design landscape.

6.2 Specific Empirical Questions

1. On long-context reasoning benchmarks (MathHay, Needle Threading, multi-hop QA over very long documents), how do pure-SSM, pure-attention, and hybrid architectures of matched parameter count compare at matched compute and memory budgets?
2. Does the optimal attention-to-SSM ratio in hybrid architectures depend on task type? A controlled study varying ratios across reasoning, retrieval, and generation workloads would settle a question that is currently answered by intuition.
3. How much of the long-context reasoning gap is architectural and how much is training-data-driven? A study that retrains the same architecture on long-context reasoning datasets versus standard data would isolate this.
4. In production hybrid models, which layer types are doing the most work for which subtasks?

Mechanistic-interpretability-style probing of Jamba or Granite 4 layers would reveal whether the attention layers are preferentially activated during reasoning operations as the design intuition assumes.

5. For agentic memory systems, what is the practical ceiling on persistent-memory size before retrieval quality and consistency degrade, and how does this depend on the underlying retrieval mechanism?

7. Conclusion

The discussion of memory architectures beyond attention has not kept pace with where the field actually stands. Hybrid attention-SSM architectures are in production, pure transformers reach 10M-token contexts with near-perfect retrieval, and the retrieval-at-long-context problem that motivated much of the SSM research programme is largely solved at the frontier. What remains genuinely open is reasoning at long context, and the four-way conflation of "memory" concepts has obscured this by allowing improvements on one axis to be reported as if they addressed all four.

The productive research agenda is not to find a single architecture that replaces attention. It is to characterise which architectural choices help with which workloads, to build hybrid designs that adaptively route compute to where it is most needed, and to develop benchmarks that distinguish the easy long-context problems from the hard ones. The field has the architectural toolkit. What it lacks is the methodological discipline to use the toolkit in ways that cumulate into reliable design knowledge.

References

- Arora, S., Eyuboglu, S., Timalsina, A., Johnson, I., Poli, M., Zou, J., Rudra, A., & Ré, C. (2023). Zoology: Measuring and improving recall in efficient language models. arXiv:2312.04927.
- Dao, T., Fu, D., Ermon, S., Rudra, A., & Ré, C. (2022). FlashAttention: Fast and memory-efficient exact attention with IO-awareness. NeurIPS 2022.
- Dao, T., & Gu, A. (2024). Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality (Mamba-2). International Conference on Machine Learning (ICML).
- Gu, A., & Dao, T. (2023). Mamba: Linear-time sequence modeling with selective state spaces. arXiv:2312.00752 (later Conference on Language Modeling).
- Gu, A., Goel, K., & Ré, C. (2022). Efficiently modeling long sequences with structured state spaces (S4). International Conference on Learning Representations (ICLR).
- Katharopoulos, A., Vyas, A., Pappas, N., & Fleuret, F. (2020). Transformers are RNNs: Fast autoregressive transformers with linear attention. International Conference on Machine Learning (ICML).
- Lieber, O., et al. (2024). Jamba: A hybrid Transformer-Mamba language model. AI21 Labs Technical Report.
- Liu, A., et al. (2026). Not all needles are found: How fact distribution and don't make it up prompts shape literal extraction, logical inference, and hallucination risks in long-context LLMs. arXiv:2601.02023.
- Peng, B., Alcaide, E., Anthony, Q., Albalak, A., Arcadinho, S., Cao, H., et al. (2023). RWKV: Reinventing RNNs for the transformer era. EMNLP.
- Reid, M., Savinov, N., Teplyashin, D., Lepikhin, D., Lillicrap, T., Alayrac, J.-B., et al. (2024). Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. Google DeepMind. arXiv:2403.05530.
- Ren, L., Liu, Y., Lu, Y., Shen, Y., Liang, C., & Chen, W. (2024). Samba: Simple hybrid state space models for efficient unlimited context language modeling. arXiv:2406.07522.
- Roberts, J., et al. (2024). Needle threading: Can LLMs follow threads through near-million-scale haystacks? arXiv:2411.05000.
- Sun, Y., Dong, L., Huang, S., Ma, S., Xia, Y., Xue, J., et al. (2023). Retentive network: A successor to transformer for large language models. arXiv:2307.08621.
- Sun, Y., Li, X., Dalal, K., Xu, J., Vikram, A., Zhang, G., et al. (2024). Learning to (learn at test time): RNNs with expressive hidden states. arXiv:2407.04620.
- Waleffe, R., et al. (2024). An empirical study of Mamba-based language models. arXiv preprint.
- Wang, L., et al. (2024). MathHay: An automated benchmark for long-context mathematical reasoning in LLMs. arXiv:2410.04698.
- Yang, S., Wang, B., Shen, Y., Panda, R., & Kim, Y. (2024). Gated linear attention transformers with hardware-efficient training. ICML 2024.