APPLIED FUTURES RESEARCH
— Whitepaper Series —

Sub-Quadratic Sparse Attention

The geometry that broke the transformer — and where it leads next

A Technical Whitepaper on Long-Context Language Model Architecture

Current State · Evolutionary Arc · Forward Trajectory · Successor Architectures

Prepared by

Kevin Nussbaum

Applied Futures Research

May 8, 2026

Scope and disclosure

This is a technical analysis of an active area in machine-learning research, prepared for educational and reference purposes. It draws on the published literature through the first half of 2026 and includes forward-looking projections clearly identified as such. Numerical figures for cost and capacity are order-of-magnitude estimates; assumptions are stated inline. Nothing in this document represents an endorsement, investment recommendation, or engineering certification.

Contents

1. The Quadratic Wall

Every modern large language model is built on a mechanism called self-attention. For each token in a sequence, the model computes a relationship score against every other token. The computation is elegant, parallelizable, and — at small scales — almost free. It is also, mathematically, a square.

If a sequence contains N tokens, the attention layer performs roughly N^2 operations. Doubling the input quadruples the work. At a thousand tokens, that figure represents a million operations per layer per head — trivial. At a million tokens, it becomes a trillion. At ten million, ten trillion. The compute curve does not bend; it accelerates upward into a wall.

Memory tracks the same curve. The attention matrix itself — the table of every-token-to-every-token scores — must be held in fast memory during inference. Long contexts have therefore been less a software problem than a thermodynamic one: the chip simply cannot hold the matrix.

This is the ceiling that has historically capped context windows, throttled long-document reasoning, and made million-token inference financially absurd. Sub-quadratic sparse attention is the engineering response to that ceiling — and increasingly, the breach in it.

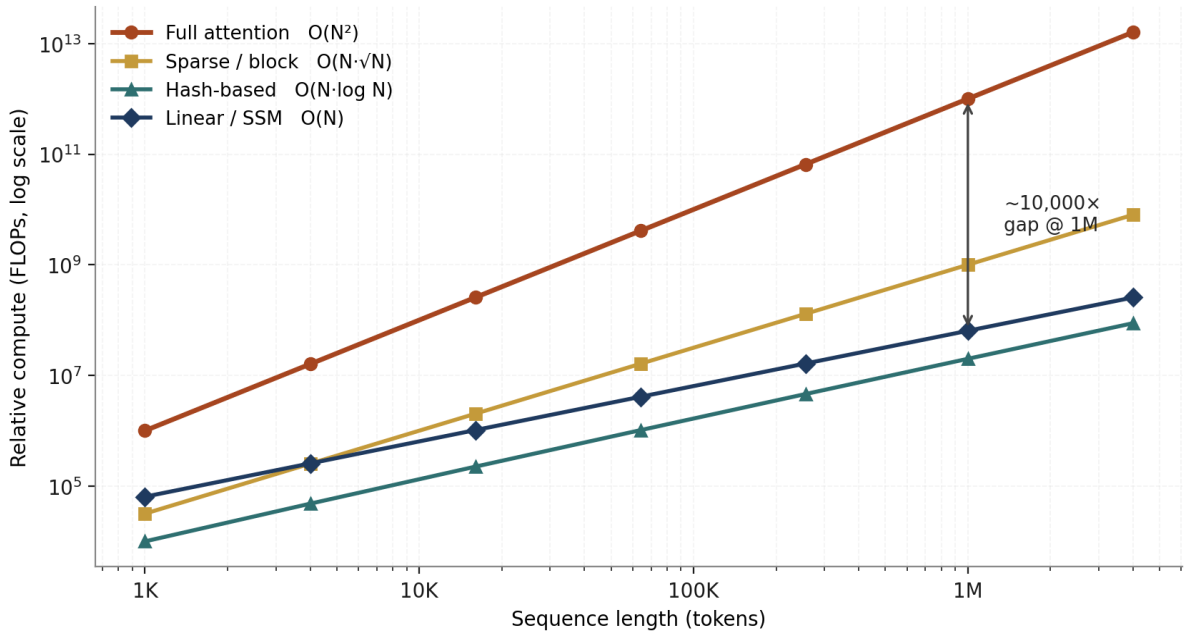

Figure 1. Compute scaling for full attention versus sub-quadratic alternatives. The vertical arrow at the one-million-token mark shows the gap between $O(N^2)$ and $O(N)$ — approximately four orders of magnitude. This is the difference between a query that costs a few cents and one that costs hundreds of dollars; in practical terms, between commercially viable and physically impossible.

2. What Sparse Attention Actually Does

Break the term apart. Sparse means each token no longer attends to every other token, but to a deliberately chosen subset. Sub-quadratic means the total compute scales slower than N^2 — typically $N \cdot \log(N)$, $N \cdot \sqrt{N}$, or near-linear N.

The intuition is plain enough. When a model processes the four-hundredth page of a novel, it does not need to re-read every word of the first three hundred and ninety-nine to understand the next sentence. Most token pairs carry no useful signal for one another. The trick is choosing which subset to keep without sacrificing the long-range connections that gave full attention its power in the first place.

2.1 Four canonical patterns

Sliding window. Each token attends only to its nearest neighbors — the last K tokens. Linear in cost. Excellent for local coherence; useless for long-range recall on its own.

Local plus global. Combine a sliding window with a small set of "global anchor" tokens that every position can see. The Longformer family (2020) and BigBird (2020) showed that this hybrid recovers most of full attention's quality at a fraction of the cost.

Hash-based routing. Tokens with similar content are bucketed together via locality-sensitive hashing, and attention runs only inside each bucket. This was Reformer's contribution (2020) and a precursor to today's mixture-of-experts routing logic.

Learned and dynamic sparsity. The 2024–2026 frontier. The model itself learns which subsets matter, often at the level of token blocks rather than individual tokens. DeepSeek's Native Sparse Attention and Kimi's Mixture-of-Block-Attention are leading exemplars. These are the methods now closing — and in some benchmarks crossing — the quality gap with full attention.

2.2 The hybrid principle

No single sparse pattern dominates. In practice, modern frontier architectures combine multiple patterns within one model: dense attention in the first few layers (where context is short and N^2 is cheap), block-sparse attention in the middle layers (where most of the compute lives), and state-space layers for the very long tail. Each component handles what it is best at. The pure transformer, monolithic and quadratic, has become a historical reference rather than a deployed architecture.

3. Where This Sits in the Evolution of the LLM

Sub-quadratic sparse attention is not a new idea. It is the latest move in a sequence of architectural decisions that has been running since the original transformer paper. The arc tells the story.

2017 · The genesis

Vaswani et al. publish “Attention Is All You Need.” Self-attention replaces recurrence as the dominant sequence-modeling primitive. The N^2 cost is acknowledged as a limitation in the very first paper but is dismissed as tolerable at then-current sequence lengths.

2018 – 2019 · The pre-training era

BERT, GPT-2, T5. Context windows hold at 512 to 1,024 tokens. The quadratic ceiling is invisible because no one is trying to push past it. The community's attention is on scale, not length.

2019 – 2021 · First sparse attempts

Sparse Transformer, Longformer, BigBird, Reformer. Researchers begin the assault on N^2. Most of these methods sacrifice quality for compute savings; production systems decline to adopt them. The era's verdict is mixed: sparse attention works, but not well enough to displace full attention where compute is available.

2022 · The kernel breakthrough

Tri Dao's FlashAttention reduces memory cost without changing the math. It is full attention — exact, dense, quadratic — but executed with an IO-aware kernel that keeps the working set inside the GPU's fast on-chip memory. Suddenly 32K, 64K, 100K-token windows are feasible with full attention. The community concludes — prematurely — that sparsity is unnecessary.

2023 · Long context goes mainstream

GPT-4, Claude 2, and the first 100K-class production windows arrive. KV-cache optimizations and FlashAttention v2 carry the load. State-space models (Mamba) appear as a credible post-transformer alternative, demonstrating linear-time sequence modeling competitive with attention on certain benchmarks.

2024 · The million-token threshold

Gemini 1.5 and Claude 3 cross 1 million tokens in production. Pure quadratic attention at that length is economically impossible — the underlying systems are already hybrid, even when the marketing does not say so. The sparse era has, quietly, begun.

2025 – 2026 · Native sparse

A new generation of methods — DeepSeek's NSA, Kimi's MoBA, hybrid stacks like Jamba and Samba — trains the sparsity pattern natively, sparse from the first weight update, rather than retrofitting sparsity onto a dense model. Quality matches or exceeds full attention on long-context benchmarks. The architecture is no longer pure transformer; it is transformer plus state-space plus learned sparse, blended layer by layer.

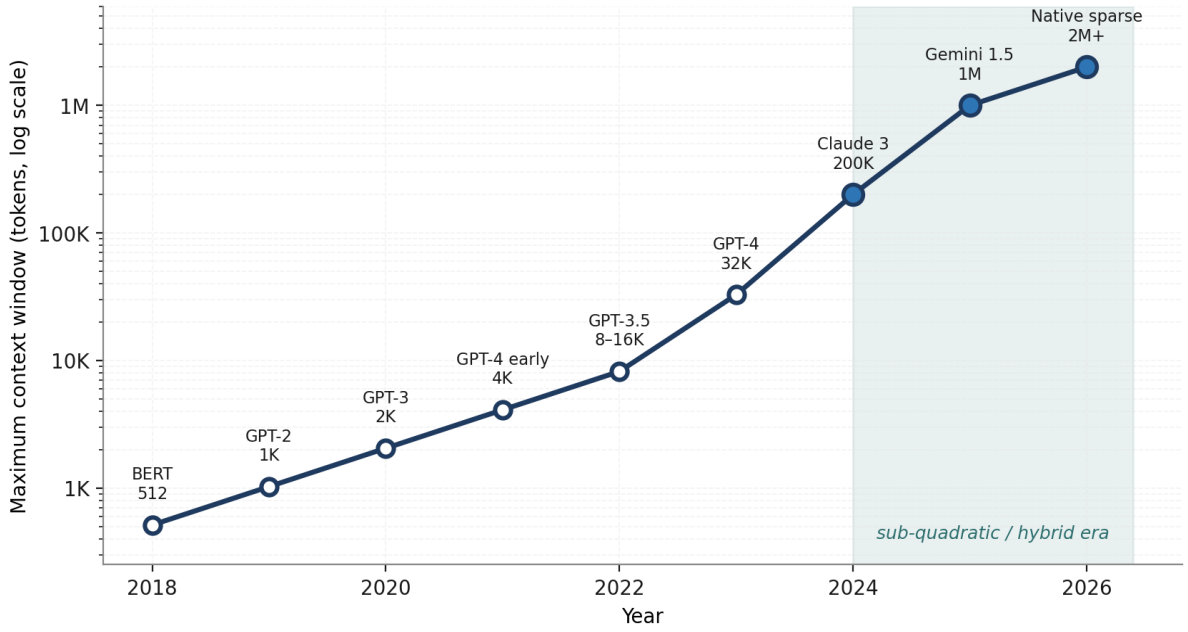

Figure 2. Frontier-model maximum context windows on a logarithmic scale, 2018 through 2026. The acceleration after 2023 is the visible signature of the sub-quadratic transition; the shaded region marks the era in which production systems stopped relying on pure full-attention to support their advertised window sizes.

The transformer of 2026 shares a name with the transformer of 2017, but inside, it is a different machine. Full quadratic attention is no longer the load-bearing element of the architecture. It is the residue.

4. What Current LLMs Gain

The benefits land at four levels of the stack at once.

4.1 Long context becomes affordable

The 200K to 2M token windows that consumers now take for granted are not the product of brute-force scaling. They are the product of sparse and hybrid attention combined with KV-cache compression, FlashAttention kernels, and mixture-of-experts routing. Without the sub-quadratic move, the economics simply do not work — the cost of inference at long context grows fast enough to outrun any plausible price the market would pay.

4.2 Prefill latency drops

When a user pastes a hundred-page document, the model must process the entire input before producing the first output token — the so-called prefill stage. Prefill is dominated by attention compute. Sparse methods cut that wait time substantially, often by an order of magnitude at long contexts. The user-perceived effect is the difference between an app that feels usable and one that feels broken.

4.3 Memory pressure relaxes

The KV cache — the stored keys and values from prior tokens, consulted at every decode step — is one of the worst memory hogs in modern inference. Sparse attention shrinks the working set the cache must serve, allowing more concurrent users per GPU and longer sustained generation per session. The economics of inference scale with this number.

4.4 Per-token cost falls

The cumulative effect on pricing has been dramatic. A million input tokens of long-document analysis cost roughly 90 percent less in 2026 than the same workload would have cost in 2023. The trajectory is consistent across providers and consistent across model classes.

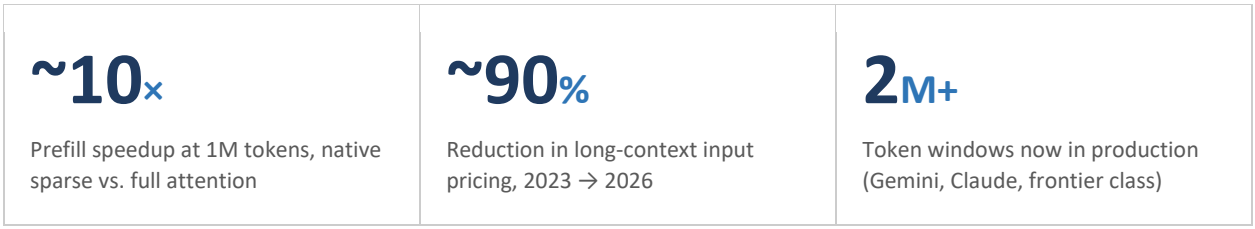

Figure 3. Three measures of the practical payoff. Prefill speedup is reported by DeepSeek and Moonshot for native sparse implementations against full-attention baselines; pricing reduction is observed across major API providers; window size reflects 2026 production deployments.

5. Where This Points

The next three to five years are not about whether sub-quadratic attention works. That question is settled. The next three to five years are about what becomes possible once it works everywhere.

5.1 Effectively unlimited context

Ten million tokens is not a stretch from where the field stands today. A hundred million is plausible by 2028. At those scales, the engineering problem ceases to be “can the model fit it?” and becomes “can the model actually use all of it well?” — a distinct, harder, currently unsolved problem that this paper returns to in Section 6.

5.2 Persistent agents

Long-running agents that maintain coherent memory of hours, days, or weeks of activity require cheap long-context inference as foundational infrastructure. Without sub-quadratic attention, an agent's memory is a gimmick — a sliding summary that loses fidelity over time. With it, the agent can hold the literal record. This is the architectural prerequisite for the agent economy that is only beginning to form in 2026.

5.3 The hybrid architecture wins

Pure sparse attention is unlikely to dominate on its own. The future, already visible in 2025–2026, is layered: local sliding-window attention for short range, learned block-sparse attention for medium range,

and state-space models (Mamba, Samba, Jamba) for the very long tail. Each handles what it is best at. The pure transformer becomes one component in a larger ensemble, not the ensemble itself.

5.4 On-device inference matures

Sub-quadratic methods translate directly into smaller memory footprints, which translate into models running on phones and laptops. Long-document reasoning on local hardware — with the privacy and latency advantages that implies — moves from research demonstration to consumer product over the next 18 to 24 months.

5.5 The cost compression continues

Long-context input pricing should fall another five to tenfold over the next two to three years. The current steep gradient between short-context and long-context pricing — where a million-token query costs vastly more per query than a thousand-token one — should compress meaningfully. Long context will stop being a premium tier and start being the default.

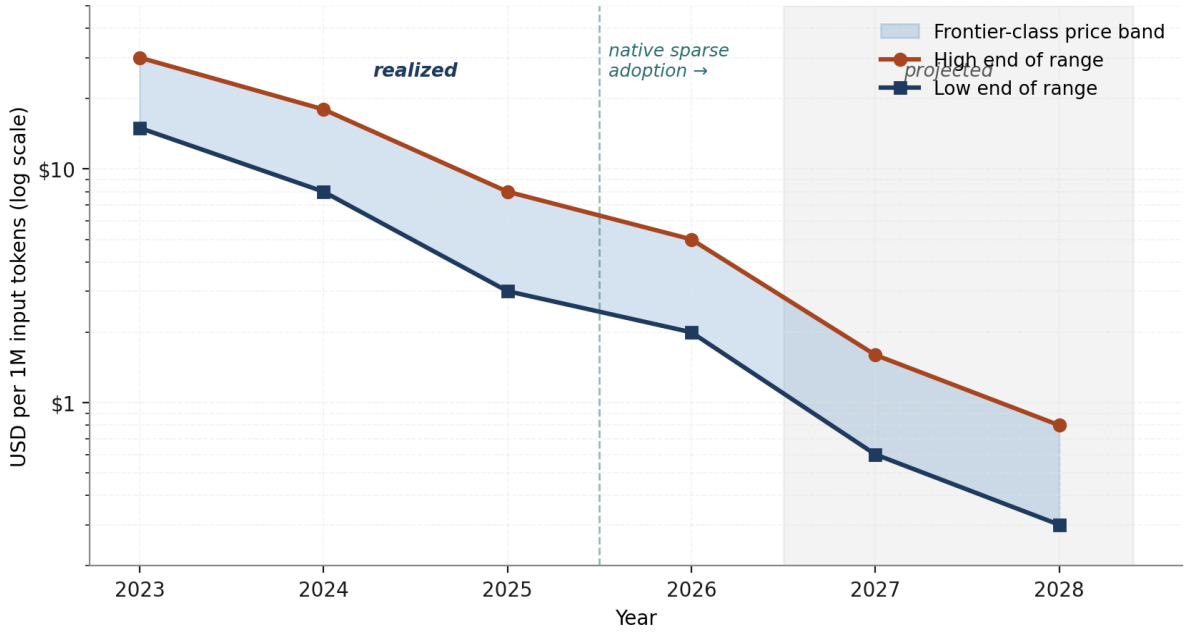

Figure 3 (continued). Long-context input pricing trajectory, USD per one million tokens, frontier-class models, 2023 through 2028. The shaded band marks the projected portion. The acceleration after 2025 reflects the broader adoption of native sparse training methods, which both reduce inference cost and allow providers to offer long context as a commodity tier rather than a premium feature.

5.6 Capability summary

Drawing the threads together, the current and projected state of long-context machine intelligence looks like this:

Capability	2023	2026	~2028 projection

Frontier context window	~32K	1M – 2M	10M – 100M
Cost per million input tokens	~$15 – $30	~$2 – $5	~$0.30 – $0.80
Dominant attention scaling	$O(N^2)$ full	$O(N \cdot \sqrt{N})$ hybrid	O(N) hybrid + SSM
Practical agent memory	minutes	hours – days	weeks – months
On-device long-context	research only	~32K on laptop	~1M on phone

Table 1. Long-context capability and economics, observed and projected, 2023 to ~2028. Cost figures are typical frontier-class API pricing for input tokens; agent memory figures reflect coherent recall under realistic prompting, not raw context window.

6. What Sub-Quadratic Attention Can Miss

Sparse methods preserve the average case. The unsettled question is whether they preserve the worst case — the rare but consequential long-range dependency that full attention catches almost by definition. The classic failure mode is the "needle in a haystack" task: a single critical fact buried at token 847,332 that the model must surface to answer correctly. Sparse patterns, by their nature, may simply not look there.

This matters more in some domains than others. For casual chat, summarization, and creative work, average-case quality is what users notice. For legal review, medical reasoning, code audit, and compliance — the high-stakes domains where AI is increasingly being deployed — the worst case is the only case that matters. A contract clause missed because it sat in the wrong sparse block is a clause missed.

The frontier labs are aware of this. Native sparse methods like NSA explicitly include occasional global passes precisely to guarantee the long-range tail is sampled. Whether those guarantees are strong enough for high-stakes applications is the open empirical question of the next two years. The honest answer, today, is: probably yes for most uses, demonstrably no for some, and the boundary is moving.

The deeper open question. We have learned how to compute over arbitrarily long sequences. We have not yet learned how to reason over them. Capacity is not comprehension — and the gap between the two is now the most important unsolved problem in the field.

7. What Comes After Sparse Attention

The honest answer is that the transformer's days as the singular dominant architecture are numbered. The signs are already visible.

State-space models — Mamba, S4, and their descendants — handle very long sequences in genuinely linear time, with a fundamentally different mathematical structure (continuous-time dynamics, selective recall) than attention. Their weakness has historically been in-context learning and certain copy-paste

tasks where attention excels. Recent work shows hybrid stacks (Jamba, Samba, Zamba) outperforming pure transformers of equivalent size on long-context benchmarks while running materially faster.

The likely successor is not a single new mechanism but an architectural composition: SSM layers for the linear-time long-range scan, sparse attention layers for the selective high-resolution lookups, and dense attention reserved for short-range work where N^2 is cheap because N is small. Each layer does what it does best.

Beyond that horizon — five to ten years — the speculation gets thinner but the direction is suggestive.

7.1 Continuous attention

Instead of a discrete attention matrix, treat memory as a continuous field that the model integrates over. Early work on neural memory, modern Hopfield networks, Titans, and continuous-time architectures gestures in this direction. The promise is a memory that does not grow with token count at all — that compresses as it accumulates.

7.2 Hierarchical context

A model with not one context window but a tree of them — paragraph-level, chapter-level, document-level — with attention operating at each scale separately and interactions across scales. The brain almost certainly does something like this; current LLMs do not. Emerging research on multi-scale state-space models is the closest active analogue.

7.3 Learned compression as native operation

Rather than storing every token's keys and values, the model continuously compresses old context into denser representations during inference itself. The KV cache becomes a working memory, not an archive. The boundary between weights and context dissolves.

7.4 What unifies them

What unifies these directions is the recognition that the attention matrix — the single most important data structure in modern AI — was always a temporary compromise. It is large, expensive, and informationally redundant. Whatever replaces it will be smaller, cheaper, and more structured. Sub-quadratic sparse attention is the bridge to that successor. It is not the destination.

8. The Geometry of the Next Decade

Every decade of computing has been defined by a particular geometric problem and the engineering that solved it. The 1990s solved memory hierarchy with caches. The 2000s solved parallelism with multi-core. The 2010s solved neural training with backpropagation at scale on GPUs. The 2020s, in its first half, has been about solving the quadratic ceiling of attention.

Sub-quadratic sparse attention is not the last word on that problem, but it is the move that converted a hard wall into a soft gradient. Context windows now grow cheaply. Inference now scales gracefully. Agents now have memory that persists across meaningful timeframes. The capability frontier of language models is no longer determined by what compute can support; it is increasingly determined by what reasoning can use.

That is the next problem, and it is a different kind of problem — less about silicon and more about cognition. Capacity has been won. Comprehension is open.

Selected References and Further Reading

- Vaswani, A. et al. (2017). Attention Is All You Need. NeurIPS.
- Child, R. et al. (2019). Generating Long Sequences with Sparse Transformers. arXiv:1904.10509.
- Beltagy, I., Peters, M. E., Cohan, A. (2020). Longformer: The Long-Document Transformer. arXiv:2004.05150.
- Zaheer, M. et al. (2020). Big Bird: Transformers for Longer Sequences. NeurIPS.
- Kitaev, N., Kaiser, Ł., Levskaya, A. (2020). Reformer: The Efficient Transformer. ICLR.
- Dao, T. et al. (2022). FlashAttention: Fast and Memory-Efficient Exact Attention with IO-Awareness. NeurIPS.
- Gu, A., Dao, T. (2023). Mamba: Linear-Time Sequence Modeling with Selective State Spaces. arXiv:2312.00752.
- Lieber, O. et al. (2024). Jamba: A Hybrid Transformer-Mamba Language Model. AI21 Labs technical report.
- DeepSeek-AI (2025). Native Sparse Attention: Hardware-Aligned and Natively Trainable Sparse Attention. arXiv preprint.
- Lu, J. et al. (2025). MoBA: Mixture of Block Attention for Long-Context LLMs. Moonshot AI / Kimi technical report.