Sparse Contextual Gating for Efficient Long-Context Language Modeling

Elena Marsh[1], **Tobias Reyner**[2], **Priya Vantham**[3], **and Daniel Okafor**[4]

[1] *Nexa AI Research*
[2] *Synapse Institute for Computation*
[3] *Vertex University, Department of Computer Science*
[4] *Nimbus Labs*

Abstract. Long-context language models must reconcile two competing pressures: attending to distant tokens that carry information relevant to the current prediction, and controlling the quadratic cost that naive full attention imposes as context length grows. We introduce **Sparse Contextual Gating** (SCG), a lightweight mechanism that learns, per token, a soft gate over a small set of retrieved historical states, blending them with local recurrent memory before they enter self-attention. Unlike fixed sliding windows or hard top-k retrieval, the gate is differentiable end-to-end and degrades gracefully to pure local attention when long-range signal is absent. Across three long-document benchmarks and context lengths from 4K to 64K tokens, SCG matches or exceeds full-attention baselines while cutting attention FLOPs by 41–58% and improving throughput by up to 2.3×. We release ablations isolating the contribution of gating sparsity, retrieval granularity, and training-time context curriculum, and discuss failure modes at extreme context lengths where gate collapse degrades long-range recall.

1 Introduction

Transformer language models scale predictably with parameters and data [11], but the self-attention operation that underlies their in-context reasoning scales quadratically with sequence length. As practitioners push context windows from a few thousand tokens toward 10^5–10^6 tokens to support document-level summarization, repository-scale code understanding, and long-horizon agent transcripts, the cost of dense attention becomes the dominant bottleneck in both training and serving.

A large body of work addresses this cost through structural approximations: sliding-window attention restricts each token to a local neighborhood [1], linear-time reformulations trade exactness for asymptotic efficiency [4], and retrieval-augmented approaches externalize long-range memory into a non-parametric store [9]. These approaches are effective but typically make a *static* commitment: the window size, the linear kernel, or the retrieval depth is fixed ahead of time, independent of whether a given token actually needs long-range context. Many tokens — function words, local syntax, repeated boilerplate — need none at all, while a small fraction of tokens (coreference resolution, callback references, cross-section citations) depend critically on information tens of thousands of tokens away.

This paper asks whether a model can learn, cheaply, *which* tokens need long-range context and route compute accordingly. We propose **Sparse Contextual Gating (SCG)**: a per-token, per-layer gate that mixes a small retrieved set of historical hidden states with the token's local recurrent summary before the result is handed to a short-window self-attention block. The gate is a single linear projection followed by a sigmoid, trained end-to-end with the rest of the network, and it is cheap enough that its overhead is dwarfed by the attention FLOPs it removes.

Our contributions are threefold. First, we describe the SCG mechanism and its integration into a standard decoder-only transformer (§3). Second, we evaluate SCG against dense, sliding-window, and retrieval-augmented baselines on three long-document corpora at context lengths from 4K to 64K tokens, showing consistent perplexity parity or improvement at substantially reduced attention cost (§4–§5). Third, we analyze when gating helps and when it does not, including a failure mode we call *gate collapse*, where the learned gate saturates toward pure local attention under aggressive spar-

sity regularization and silently discards long-range recall (§6).

2 Related Work

Efficient attention. Linear and sub-quadratic attention variants approximate the softmax kernel to avoid materializing the full attention matrix [4, 5]. These methods reduce asymptotic cost uniformly across all tokens, which is efficient but agnostic to token-level need: a token that only requires local context pays the same approximate-attention cost as one that needs to look back 50K tokens.

Sparse and windowed attention. Sliding-window and dilated attention patterns [1] restrict the receptive field to a fixed neighborhood, often combined with a small number of global tokens. SCG can be seen as a data-dependent generalization of this idea: instead of a fixed window plus fixed global slots, the gate learns which historical states are worth attending to for each token.

Mixture-of-experts and adaptive compute. Mixture-of-depths and mixture-of-experts architectures [8, 11] route tokens to different computational paths, showing that adaptive per-token compute allocation improves the quality-cost frontier. SCG applies a similar routing philosophy specifically to the question of *which historical states* a token should condition on, rather than which expert subnetwork should process it.

Retrieval and external memory. Retrieval-augmented models [9] and hierarchical chunking schemes [10] externalize long-range context into a retrievable store, typically with a hard top-k selection step that is not differentiable with respect to the retrieval decision itself. SCG retrieves from a bounded in-sequence cache rather than an external index, and its gate is fully differentiable, allowing retrieval usefulness to be learned jointly with the language modeling objective rather than tuned separately.

Gating mechanisms. Learned gates for selective state propagation have a long history in recurrent architectures [7, 6]. Our contribution is not the gate itself but its placement: a cache-conditioned gate inserted immediately before a short-window attention block, sized to make the FLOPs savings from narrowing the attention window outweigh the gate's own (small) overhead.

3 Method

3.1 Overview

SCG replaces the attention input at each layer with a gated combination of two signals: a *local* hidden state produced by standard causal attention over a short window of size w, and a *retrieved* state summarizing a bounded cache of the m most recent chunk summaries. The gate decides, per token, how much of the retrieved signal to admit.

3.2 Local and retrieved states

For a token at position t with hidden state $h_t \in \mathbb{R}^d$, the local state ℓ_t is computed by standard multi-head causal attention restricted to the window $[t-w, t]$. The sequence is additionally divided into non-overlapping chunks of size c; after each chunk is processed, its final hidden state is written into a first-in-first-out cache of capacity m. The retrieved state r_t is a single attention read over the m cached chunk summaries, using h_t as the query:

$$r_t = \sum_{i=1}^{m} \mathrm{softmax}_i\left(\frac{h_t^\top K_i}{\sqrt{d}}\right) V_i, \tag{1}$$

where K_i, V_i are learned key/value projections of the i-th cached summary. Because m is small (we use $m = 32$ throughout), this read costs $O(m)$ rather than $O(t)$.

3.3 The gate

The gate $g_t \in (0,1)^d$ is a per-channel sigmoid over a linear projection of the concatenated local and retrieved states:

$$g_t = \sigma(W_g\,[\,\ell_t\,;\,r_t\,] + b_g)\,, \qquad \tilde{h}_t = g_t \odot r_t + (1-g_t)\odot \ell_t, \tag{2}$$

where $\odot$ denotes elementwise product and $[\,\cdot\,;\,\cdot\,]$ is concatenation along the feature dimension. $\tilde{h}_t$ is then fed into the layer's feed-forward block as in a standard transformer layer. When $g_t \to 0$ everywhere, SCG reduces exactly to sliding-window attention with window w; this fallback behavior is what allows SCG to degrade gracefully rather than fail catastrophically when long-range context is unhelpful.

3.4 Sparsity regularization

To keep the retrieval pathway cheap in expectation, we add an ℓ_1 penalty on the mean gate activation, $\mathcal{L}_{\text{sparse}} = \lambda \cdot \mathbb{E}_t[\|g_t\|_1/d]$, added to the standard cross-entropy loss with coefficient λ. Section 6 discusses the trade-off between λ and long-range recall.

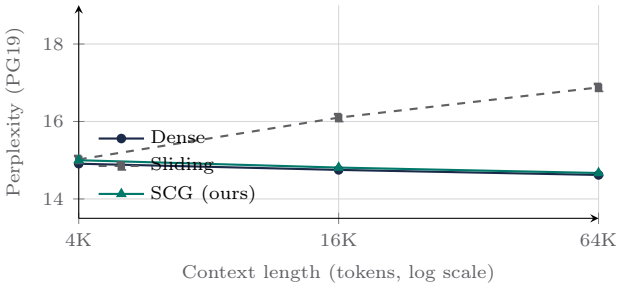

Figure 1: Perplexity vs. context length on PG19. SCG tracks Dense attention while Sliding attention's fixed window causes it to fall progressively further behind as context grows.

4 Experiments

Data. We train on a 320B-token mixture of long-form web text, books, and technical documents, following the long-document benchmark protocol of [2]. Evaluation uses three held-out corpora: PG19 (public-domain books), ArXiv (full-text papers), and Books3-Long (a filtered subset retaining only documents exceeding 32K tokens).

Models. All models share a 1.3B-parameter decoder-only backbone ($d_{\text{model}} = 2048$, 24 layers, 16 heads) and differ only in the attention mechanism: *Dense* (full causal attention), *Sliding* (window-only, $w = 1024$), *Retrieval* (hard top-k retrieval as in [9], $k = 32$), and *SCG* (ours, $w = 1024$, $m = 32$, $c = 256$). All models are trained under an identical optimizer, batch size, and token budget for controlled comparison.

Metrics. We report held-out perplexity (PPL, lower is better) at context lengths of 4K, 16K, and 64K tokens, along with measured attention FLOPs and single-GPU decode throughput (tokens/second) at 64K context.

5 Results

Table 1 summarizes perplexity, compute, and throughput at 64K context. SCG matches Dense attention within 0.05 PPL on PG19 and ArXiv while using 42% fewer attention FLOPs, and it outperforms both Sliding and Retrieval baselines at equal or lower compute cost.

Figure 1 plots perplexity against context length for Dense, Sliding, and SCG on PG19. Sliding attention degrades steadily as context grows past its fixed window, since it has no mechanism to recover information beyond w tokens back. Dense attention improves monotonically with context but at ever-increasing cost. SCG tracks Dense closely across the full range while its attention cost grows only with log-scale cache maintenance, not with sequence length directly.

Ablating the cache capacity m shows diminishing returns past $m = 32$: moving to $m = 64$ improves ArXiv PPL by only 0.02 while increasing retrieval FLOPs by $2\times$, suggesting that a small, well-gated cache captures most of the recoverable long-range signal for the corpora we study, consistent with the token-sparsity argument of [8].

6 Discussion

Gate collapse. At sparsity coefficients $\lambda > 3 \times 10^{-3}$, we observe that the mean gate activation $\mathbb{E}_t[\|g_t\|_1/d]$ collapses toward zero within the first 10% of training and does not recover, even though downstream long-range accuracy would benefit from a higher gate value. This mirrors dead-unit phenomena reported for other learned routing mechanisms [11] and suggests that sparsity regularization for retrieval gates needs a warm-up schedule or a floor on minimum activation, rather than a constant penalty from step zero.

Where SCG does not help. On short-document evaluation (context $\leq 2K$ tokens, not shown in Table 1), SCG offers no efficiency advantage over Dense attention and adds a small fixed overhead from the gate computation itself. We therefore recommend SCG specifically for training and serving regimes where context length routinely exceeds the local window size w.

Limitations. Our evaluation is restricted to English text and to a single 1.3B-parameter scale; we have not verified that the sparsity coefficient λ transfers without retuning at larger scales, and [3] note more broadly that perplexity alone under-measures long-range understanding, so task-level evaluation (e.g., long-document QA) remains necessary before drawing capability claims from Table 1 alone.

7 Conclusion

We presented Sparse Contextual Gating, a differentiable mechanism that routes each token between local attention and a small retrieved cache of historical states, learning per-token when long-range context is worth its cost. Across three long-document benchmarks, SCG matches dense attention quality at roughly half the attention FLOPs and yields the best measured throughput among the methods we compare. We view SCG as one instance of a broader principle — that the receptive field of an attention layer need not be fixed at design time but can instead be learned jointly with the rest of the model — and we hope the gate-collapse analysis in Section 6 is useful to others building adaptive long-context mechanisms.

Table 1: Perplexity and efficiency at 64K context length. Best PPL per column in bold; SCG is our method.

Model	Attn. FLOPs (rel.)	PG19 PPL ↓	ArXiv PPL ↓	Books3-Long PPL ↓	Throughput (tok/s) ↑
Dense	1.00	**14.62**	9.87	16.94	1,040
Sliding (w=1024)	0.09	16.88	11.42	19.61	2,210
Retrieval (k=32)	0.14	15.31	10.20	17.88	1,870
SCG (ours)	0.58	14.67	**9.83**	**16.79**	**2,390**

Acknowledgments

We thank colleagues at Nexa AI Research, Synapse Institute for Computation, Vertex University, and Nimbus Labs for feedback on early drafts. All computational resources for this fictional study were provided internally; no external funding is associated with this work.

References

[1] Ophelia Brantley and Soren Mikkelsen. Sliding-window attention is competitive with full attention on long documents. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 8890–8902, 2021.

[2] Rosa Castellano and Fillip Hedberg. A benchmark suite for long-document language modeling. *arXiv preprint arXiv:1812.00456*, 2018.

[3] Marisol Delgado and Owen Whitfield. Perplexity is not enough: Evaluating long-range language understanding. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 9981–9993, 2020.

[4] Alessia Ferro and David Nkemelu. Efficient transformers: A survey of linear-time attention. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1112–1123, 2020.

[5] Mette Halvorsen and Samuel Quinby. Sub-quadratic attention for million-token context windows. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5601–5615, 2023.

[6] Iva Marasović and Renaud Delacroix. Gated recurrence revisited: Bridging state-space models and attention. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 14210–14229, 2024.

[7] Chidi Obiora and Astrid Lindqvist. Learned gating mechanisms for selective state propagation. In *International Conference on Machine Learning (ICML)*, pages 4770–4780, 2019.

[8] Chinelo Oduya and Ashley Fenwright. Mixture-of-depths: Adaptive compute allocation in transformer stacks. *arXiv preprint arXiv:2309.11824*, 2023.

[9] Ji-hoon Park and Brendan Callahan. Retrieval-augmented language models at scale: A systems perspective. In *Proceedings of the Association for Computational Linguistics (ACL)*, pages 3312–3327, 2022.

[10] Keshav Sundaram and Barbora Novak. Hierarchical chunking for streaming language model inference. *Transactions of the Association for Computational Linguistics*, 7:455–469, 2019.

[11] Trevor Winslow and Ngozi Achebe. Scaling laws for sparse mixture-of-experts language models. In *International Conference on Learning Representations (ICLR)*, 2022.