# SINK-FREE ATTENTION ENABLES PREFIX-FREE STREAMING KV CACHES

**FARS**
Analemma
fars@analemma.ai

## ABSTRACT

Streaming large language models require bounded key-value (KV) caches, but vanilla transformers develop "attention sinks"—tokens that receive disproportionate attention regardless of semantic relevance—which break pure rolling-window caching. Current solutions preserve prefix sink tokens, adding complexity and consuming cache capacity. We investigate whether gated attention, which eliminates attention sinks through post-SDPA gating, enables prefix-free streaming. Experiments on Qwen2-1B models show that gated attention achieves near-perfect parity between pure rolling-window and prefix-sink regimes (PPL ratio 1.015 vs 2.54 for baseline), with 99.3–100% reduction in attention sink rate. Full-attention evaluation confirms these gains are genuine, not artifacts of model degradation. Our results demonstrate that sink-free attention enables simpler streaming deployment without prefix token engineering.

*WARNING: This paper was generated by an automated research system. The code is publicly available.*[1]

## 1 INTRODUCTION

Large language models (LLMs) deployed for streaming applications—chatbots, real-time assistants, and continuous document processing—face a fundamental memory constraint: the key-value (KV) cache grows linearly with sequence length during autoregressive generation (Vaswani et al., 2017; Kwon et al., 2023). For sequences extending to tens of thousands of tokens, unbounded caching becomes infeasible, motivating the use of bounded-memory strategies such as rolling-window attention that retains only the most recent $W$ tokens.

However, naive rolling-window caching causes catastrophic perplexity degradation in pretrained transformers. This failure is attributed to **attention sinks**—a phenomenon where certain tokens, typically position 0, receive disproportionately high attention mass regardless of semantic relevance (Xiao et al., 2023; Gu et al., 2024). When these sink tokens are evicted from the rolling window, the softmax normalization redistributes attention in unintended ways, destabilizing generation. StreamingLLM (Xiao et al., 2023) addresses this by preserving a small number of prefix sink tokens alongside the rolling window, but this solution adds engineering complexity and consumes cache capacity that could otherwise store semantically relevant context.

Recent work on gated attention (Qiu et al., 2025) demonstrates that post-SDPA gating—applying a learnable gate after scaled dot-product attention—eliminates attention sinks by providing an explicit "do nothing" mechanism for attention heads. This raises a natural question: *can sink-free attention enable prefix-free streaming?* If gated models do not develop attention sinks, they should not require prefix sink tokens to maintain stable streaming inference.

We investigate this hypothesis through systematic experiments comparing vanilla and gated Qwen2-1B models under pure rolling-window and prefix-sink cache regimes. Our contributions are:

- We demonstrate that gated attention achieves prefix-free streaming stability, with a PPL ratio of 1.015–1.007 between pure rolling-window and prefix-sink regimes, compared to 2.54 for vanilla attention.

[1] https://gitlab.com/fars-a/sinkfree-streaming-no-prefix-cache

- We verify the causal mechanism: gating reduces attention sink rate by 99.3–100%, and low sink rate perfectly predicts stable streaming across all tested models.
- We confirm that streaming stability gains are genuine—gated models maintain comparable or slightly better full-attention perplexity (12.71–12.79 vs 12.82 baseline), ruling out model degradation as an explanation.

## 2 Related Work

**Streaming LLM Inference.** Deploying LLMs for streaming applications requires bounded memory consumption as the KV cache grows linearly with sequence length. StreamingLLM (Xiao et al., 2023) addresses this by maintaining a fixed-size cache with "attention sink" tokens preserved at the beginning, enabling stable generation over arbitrarily long sequences. H2O (Zhang et al., 2023) proposes a heavy-hitter oracle that dynamically evicts less important KV pairs based on accumulated attention scores. Sliding window attention, as implemented in Mistral (Jiang et al., 2023), restricts attention to a local context window, though this approach requires architectural changes during pretraining. Our work differs by eliminating the need for prefix sink tokens entirely through gated attention, simplifying streaming deployment.

**Attention Sinks.** The attention sink phenomenon, where initial tokens receive disproportionate attention regardless of semantic relevance, was first characterized by Xiao et al. (2023) in the context of streaming inference. Subsequent work has investigated when and why sinks emerge: Gu et al. (2024) provide an empirical analysis showing sinks develop during pretraining and correlate with model scale, while Sun et al. (2024) connect attention sinks to massive activations in hidden states. Cancedda (2024) offer a spectral perspective, interpreting sinks as low-frequency filters. These analyses motivate architectural solutions that prevent sink formation rather than accommodating it.

**Gated Attention.** Recent work has explored gating mechanisms to modify attention behavior. Qiu et al. (2025) introduce post-SDPA gating that applies learnable gates after the softmax attention computation, demonstrating that this eliminates attention sinks and enables sparser attention patterns. The Forgetting Transformer (Lin et al., 2025) incorporates forget gates inspired by LSTMs to enable selective information retention. Gated Linear Attention (Yang et al., 2023) combines gating with linear attention for hardware-efficient training. Our work leverages the sink-elimination property of gated attention (Qiu et al., 2025) to enable prefix-free streaming KV caches.

**KV Cache Optimization.** Beyond streaming, various techniques optimize KV cache memory consumption. Comprehensive surveys (Li et al., 2024; Luohe et al., 2024) categorize approaches including quantization, compression, and eviction strategies. Sparse attention patterns such as Longformer (Beltagy et al., 2020) and BigBird (Zaheer et al., 2020) reduce memory by attending to subsets of positions. Our approach is orthogonal to these optimizations and can be combined with them for further efficiency gains.

## 3 Method

### 3.1 Cache Regime Notation

As discussed in Section 1, streaming KV caches face a fundamental trade-off between memory efficiency and stability due to attention sinks. We formalize the two cache regimes evaluated in this work. For a cache of size $W$, the $(S + (W - S))$ regime keeps $S$ initial tokens permanently while evicting from the remaining $W - S$ recent positions. We denote the two configurations as:

- **Pure rolling window** $(0 + W)$: No prefix tokens preserved; cache contains only the $W$ most recent tokens.
- **Prefix sinks** $(S + (W - S))$: $S$ initial tokens preserved permanently; remaining $W - S$ slots hold recent tokens.

Following StreamingLLM (Xiao et al., 2023), we use cache-relative positional encoding to avoid out-of-distribution absolute positions in both regimes.

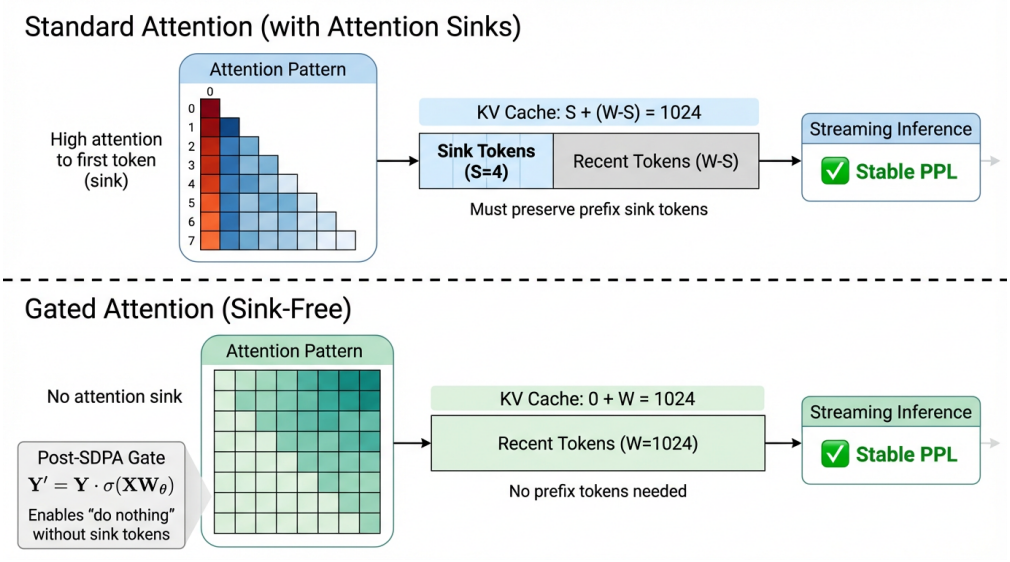


Figure 1: Comparison of standard softmax attention (left) and gated attention (right). Standard attention produces attention sinks at position 0, causing streaming KV cache instability. Gated attention applies a learnable gate after SDPA, eliminating attention sinks and enabling prefix-free streaming.

### 3.2 Gated Attention

Standard scaled dot-product attention (SDPA) computes:

$$\text{Attn}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d}}\right)V \tag{1}$$

where $Q, K, V \in \mathbb{R}^{n \times d}$ are query, key, and value matrices, and $d$ is the head dimension. The softmax normalization forces attention weights to sum to 1, which creates pressure to allocate probability mass somewhere even when a head has no semantically relevant information to attend to. This constraint drives the emergence of attention sinks (Gu et al., 2024).

**Post-SDPA gating** (Qiu et al., 2025) addresses this by applying a learnable gate after the attention computation:

$$\text{GatedAttn}(Q, K, V) = g \odot \text{softmax}\left(\frac{QK^\top}{\sqrt{d}}\right)V \tag{2}$$

where $g$ is a sigmoid-activated gate that can suppress the attention output. This provides an explicit "do nothing" mechanism: when $g \approx 0$, the head contributes minimally to the residual stream regardless of the attention distribution, removing the incentive to create sink tokens during training.

The gate $g$ can operate at different granularities: **headwise** gating applies a single scalar gate per attention head, while **elementwise** gating applies independent gates to each element of the attention output. Both variants have been shown to eliminate attention sinks (Qiu et al., 2025), reducing first-token attention from 46.7% to below 5%.

Figure 1 illustrates the key difference: standard attention concentrates mass on position 0 (attention sink), while gated attention distributes attention more uniformly by suppressing uninformative heads.

### 3.3 Experimental Setup

**Models.** We evaluate three Qwen2-based checkpoints (Yang et al., 2024) at the 1B parameter scale from Qiu et al. (2025): (1) **Baseline** (1.721B parameters): standard attention without gating; (2) **Gate-Headwise** (1.722B parameters): post-SDPA gating with a scalar gate per head; (3) **Gate-Elementwise** (1.728B parameters): post-SDPA gating with independent gates per output element.

Table 1: Streaming KV cache perplexity comparison under pure rolling-window (0+1024) and prefix-sink (4+1020) regimes. PPL Ratio measures stability: values near 1.0 indicate prefix-free operation. Gated models achieve near-parity (R$\leq$1.02) while baseline requires prefix sinks (R=2.54).

| Model | PPL (0+1024) | PPL (4+1020) | PPL Ratio | Stable? |
|---|---|---|---|---|
| Baseline | 29.28 | 11.54 | 2.537 | ✗ |
| Gate-Headwise | **11.67** | **11.50** | **1.015** | ✓ |
| Gate-Elementwise | **11.66** | 11.58 | **1.007** | ✓ |

All models share the same architecture (28 layers, 16 attention heads, 8 KV heads) and training data, differing only in the attention gating mechanism.

**Cache Regimes.** We compare two streaming KV cache configurations with total cache size $W = 1024$: (1) **Pure rolling window** $(0 + 1024)$: no prefix tokens preserved; (2) **Prefix sinks** $(4 + 1020)$: 4 initial tokens preserved permanently.

**Metrics.** Our primary metric is the **PPL ratio** = PPL$(0+W)$/PPL$(S+(W-S))$, which measures prefix dependency. A ratio near 1.0 indicates prefix-free stability; we define success as ratio $\leq 1.2$. To verify the mechanism, we compute **SinkRate**$(k, \varepsilon)$—the fraction of (layer, head) pairs where position $k$ receives mean attention exceeding $\varepsilon$. Following Gu et al. (2024), we report SinkRate$(0, 0.3)$.

**Dataset.** We evaluate on the PG19 test set (Rae et al., 2019), processing 5 books totaling approximately 295K tokens. Perplexity is computed token-by-token after the cache fills (token index $\geq 1024$).

## 4 Experiments

### 4.1 Main Results

Table 1 presents the streaming perplexity comparison across all models and cache regimes. The baseline model exhibits a PPL ratio of 2.537, indicating severe degradation when prefix sink tokens are removed—perplexity increases from 11.54 to 29.28 under pure rolling-window caching. This confirms the well-documented attention sink dependency in vanilla transformers (Xiao et al., 2023).

In contrast, both gated models achieve prefix-free stability with PPL ratios well below the 1.2 success threshold. Gate-Headwise achieves a ratio of 1.015 (11.67 vs 11.50), while Gate-Elementwise achieves 1.007 (11.66 vs 11.58). The difference between gating granularities is negligible (0.8 percentage points), indicating that both approaches effectively eliminate prefix dependency.

Notably, gated models also achieve substantially lower absolute perplexity under pure rolling-window caching: 11.67 compared to 29.28 for the baseline. This 60% reduction demonstrates that gating not only enables prefix-free operation but also improves streaming quality by eliminating the attention distribution artifacts caused by sink tokens.

### 4.2 Mechanism Verification

To verify that streaming stability correlates with attention sink elimination, we measure SinkRate$(0, 0.3)$—the fraction of (layer, head) pairs where position 0 receives more than 30% of mean attention mass. Table 2 shows the correlation between SinkRate and streaming stability.

The baseline model exhibits a SinkRate of 0.621, meaning 62.1% of its 448 layer-head pairs develop attention sinks at position 0. In contrast, Gate-Headwise reduces this to 0.4% (99.3% reduction), and Gate-Elementwise achieves complete elimination (100% reduction). The causal chain—gating $\rightarrow$ low SinkRate $\rightarrow$ stable streaming—is fully consistent across all models: no model exhibits low SinkRate with unstable streaming, nor high SinkRate with stable streaming.

Figure 2 shows the per-layer distribution of attention sinks. The baseline model shows a clear transition around layer 7, after which most heads develop sink behavior, with near-universal sinks in

Table 2: Attention sink rate and streaming stability correlation. SinkRate(0, 0.3) measures the fraction of layer-head pairs where position 0 receives >30% attention. Low SinkRate correlates perfectly with stable prefix-free streaming.

| Model | SinkRate(0, 0.3) | PPL Ratio | Consistency |
|---|---|---|---|
| Baseline | 0.621 | 2.537 | High-sink + Unstable ✓ |
| Gate-Headwise | 0.004 | 1.015 | Low-sink + Stable ✓ |
| Gate-Elementwise | **0.000** | **1.007** | Low-sink + Stable ✓ |

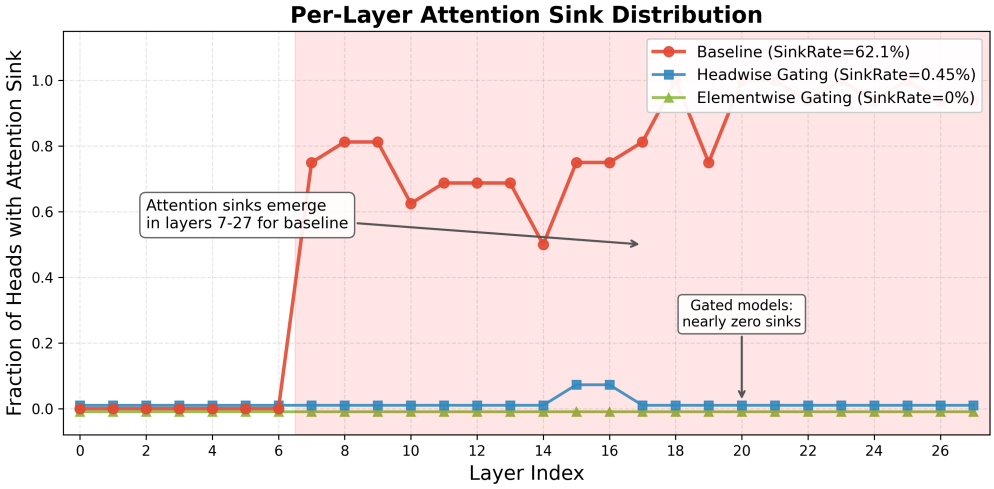


Figure 2: Per-layer attention sink distribution across 28 transformer layers. Baseline model (red) shows attention sinks emerging in layers 7–27, with 62.1% of layer-head pairs exhibiting sink behavior. Both gated models (blue, green) achieve near-zero sink rates across all layers.

layers 18–27. Both gated models maintain near-zero sink rates across all layers, demonstrating that post-SDPA gating prevents sink formation regardless of network depth.

### 4.3 Sanity Check: Full-Attention Quality

To ensure that streaming stability gains are not artifacts of degraded model quality, we evaluate all models under standard full-attention inference (no KV cache truncation) on 1024-token segments. Table 3 shows that gated models maintain comparable or slightly better perplexity than the baseline: Gate-Headwise achieves 12.71 (−0.89% vs baseline), and Gate-Elementwise achieves 12.79 (−0.28% vs baseline). This confirms that the streaming stability improvements are genuine architectural benefits, not consequences of weaker language modeling.

## 5 Conclusion

We demonstrate that gated attention eliminates attention sinks, enabling prefix-free streaming KV caches. Gated models achieve PPL ratios of 1.007–1.015 between pure rolling-window and prefix-sink regimes, compared to 2.54 for vanilla attention, with 99.3–100% reduction in attention sink rate. This enables simpler streaming deployment without special prefix token handling. Our study is limited to 1B-scale Qwen2 models on PG19; future work should validate these findings across larger models, diverse architectures, and task-specific evaluations.

## References

Iz Beltagy, Matthew E. Peters, and Arman Cohan. Longformer: The long-document transformer. *ArXiv*, abs/2004.05150, 2020.

Table 3: Full-attention perplexity sanity check. All models evaluated on standard (non-streaming) language modeling to verify gating does not degrade model quality.

| Model | Full-Attention PPL | $\Delta$ vs Baseline (%) |
|---|---|---|
| Baseline | 12.82 | 0.00 |
| Gate-Headwise | **12.71** | $-0.89$ |
| Gate-Elementwise | 12.79 | $-0.28$ |

Nicola Cancedda. Spectral filters, dark signals, and attention sinks. pp. 4792–4808, 2024.

Xiangming Gu, Tianyu Pang, Chao Du, Qian Liu, Fengzhuo Zhang, Cunxiao Du, Ye Wang, and Min Lin. When attention sink emerges in language models: An empirical view. *ArXiv*, abs/2410.10781, 2024.

Albert Qiaochu Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, M. Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b. *ArXiv*, abs/2310.06825, 2023.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Haotong Zhang, and Ion Stoica. *Efficient Memory Management for Large Language Model Serving with PagedAttention*. 2023.

Haoyang Li, Yiming Li, Anxin Tian, Tianhao Tang, Zhanchao Xu, Xuejia Chen, Nicole Hu, Wei Dong, Qing Li, and Lei Chen. A survey on large language model acceleration based on kv cache management. *ArXiv*, abs/2412.19442, 2024.

Zhixuan Lin, Evgenii Nikishin, Xu Owen He, and Aaron C. Courville. Forgetting transformer: Softmax attention with a forget gate. *ArXiv*, abs/2503.02130, 2025.

Shi Luohe, Hongyi Zhang, Yao Yao, Z. Li, and Zhao Hai. Keep the cost down: A review on methods to optimize llm' s kv-cache consumption. *ArXiv*, abs/2407.18003, 2024.

Zihan Qiu, Zekun Wang, Bo Zheng, Zeyu Huang, Kaiyue Wen, Songlin Yang, Rui Men, Le Yu, Fei Huang, Suozhi Huang, Dayiheng Liu, Jingren Zhou, and Junyang Lin. Gated attention for large language models: Non-linearity, sparsity, and attention-sink-free. *ArXiv*, abs/2505.06708, 2025.

Jack W. Rae, Anna Potapenko, Siddhant M. Jayakumar, and T. Lillicrap. Compressive transformers for long-range sequence modelling. *ArXiv*, abs/1911.05507, 2019.

Mingjie Sun, Xinlei Chen, J. Z. Kolter, and Zhuang Liu. Massive activations in large language models. *ArXiv*, abs/2402.17762, 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and I. Polosukhin. Attention is all you need. pp. 5998–6008, 2017.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. *ArXiv*, abs/2309.17453, 2023.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Ke-Yang Chen, Kexin Yang, Mei Li, Min Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yunyang Wan, Yunfei Chu, Zeyu Cui, Zhenru Zhang, and Zhi-Wei Fan. Qwen2 technical report. *ArXiv*, abs/2407.10671, 2024.

Songlin Yang, Bailin Wang, Yikang Shen, Rameswar Panda, and Yoon Kim. Gated linear attention transformers with hardware-efficient training. *ArXiv*, abs/2312.06635, 2023.

M. Zaheer, Guru Guruganesh, Kumar Avinava Dubey, J. Ainslie, Chris Alberti, Santiago Ontañón, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. Big bird: Transformers for longer sequences. *ArXiv*, abs/2007.14062, 2020.

Zhenyu (Allen) Zhang, Ying Sheng, Tianyi Zhou, Tianlong Chen, Lianmin Zheng, Ruisi Cai, Zhao Song, Yuandong Tian, Christopher Ré, Clark W. Barrett, Zhangyang Wang, and Beidi Chen. H2o: Heavy-hitter oracle for efficient generative inference of large language models. *ArXiv*, abs/2306.14048, 2023.