WHAT MAKES POSITION ZERO SPECIAL? A MECHANISTIC STUDY OF POSITION ZERO ATTENTION SINKS IN LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Transformers frequently allocate disproportionate attention to specific tokens, a phenomenon known as **attention sinks**. Causal large language models reliably form one at **position zero**, though its role remains debated. We approach this question from a mechanistic perspective, tracing how the position-zero sink arises from the model's internal computation. We identify a two-block subnetwork responsible for this behavior, which we term the **P0-Sink Circuit**, and show it arises purely from the structural properties of causal attention, requiring no semantic content. We further validate through from-scratch pre-training experiments that two proposed parameter-free methods effectively accelerate P0 sink formation, and find that earlier sink formation benefits pre-training and improves downstream performance. Both methods outperform the Transformer baseline and achieve performance comparable to Gated Attention across comprehensive settings. Code is available now at https://anonymous.4open.science/r/flash-linear-attention-6220/.

1 INTRODUCTION

Large-scale auto-regressive language models tend to allocate disproportionately large attention to a small subset of tokens, a phenomenon known as the **attention sink**. While sinks can emerge at any position, those away from the start of the input are generally harmful (Zhang et al., 2025; Yu et al., 2024): excessive focus on semantically vacuous tokens such as punctuation interferes with reasoning, and suppressing them has been shown to improve downstream performance. The **position-zero (P0) sink**, however, correlates with improved predictions and underlies several downstream applications (Xiao et al., 2024; Han et al., 2024; Chen et al., 2024). Nevertheless, recent work questions whether the P0 sink is strictly necessary, proposing architectural alternatives such as Gated Attention (Qiu et al., 2025) that claim improved downstream performance without it.

To assess the P0 sink's role, we first investigate its underlying mechanism. Prior work suggests that removing `[BOS]` leaves the sink largely intact (Gu et al., 2025), but this view is incomplete: removing `[BOS]` affects only the earliest-layer attention patterns, suggesting the sink arises from a more fundamental mechanism rooted in model architecture rather than token semantics.

From the perspective of mechanistic interpretability (Elhage et al., 2021), we reverse-engineer the Transformer and propose the **P0-Sink Circuit**, a two-block subnetwork that exploits the asymmetry of the causal attention mask. Under causal masking, position zero attends only to itself, producing an unmixed and directionally stable attention output. Subsequent MLP sublayers detect and amplify this signal into a high-norm, fixed-direction representation at position zero, which reliably triggers the P0 sink throughout the network. This explains the sink's consistent emergence across models and inputs.

Building on this mechanistic understanding, we propose two parameter-free pre-training methods that help models form the P0 sink more effectively. Our view is that the Transformer naturally favors routing attention through a consistent sink, and position zero is selected for this role precisely because it is always in context. Besides, the two functions of the P0-Sink Circuit, identifying position zero and suppressing semantic content, can be fulfilled by alternative mechanisms.

From the first perspective, we propose **Transformer-Native LM (TNLM) Loss**, which removes the cross-entropy term at the `[BOS]` position, eliminating the semantic component of the P0 repre-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

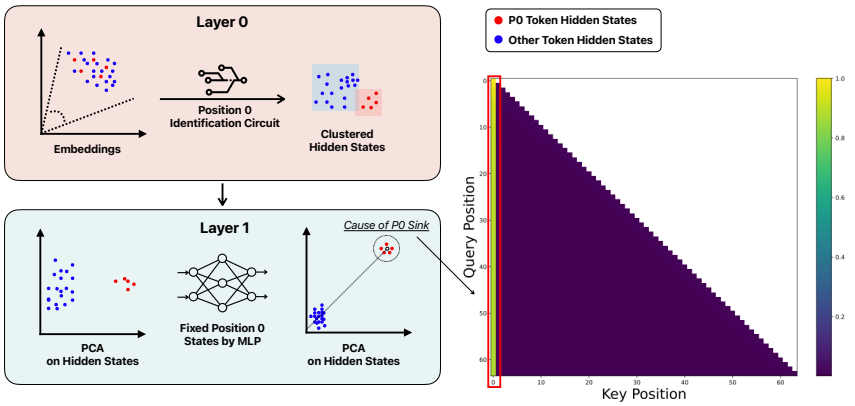


Figure 1: Overview of the proposed **P0-Sink Circuit**. Within just two transformer blocks, the model learns to identify the position-zero (P0) token and amplify it into a fixed high-norm representation, which gives rise to the attention sink effect.

sentation and freeing it to act as a dedicated attention register, inspired by register theory in vision transformer research (Darcet et al., 2024; Jiang et al., 2025). From the second perspective, we propose **Force-sink Masking**, which applies a fixed binary mask to the residual stream at every layer, hard-partitioning the hidden dimensions between position zero and all other positions. This structural partition simultaneously realizes both functions of the P0-Sink Circuit without additional parameters, and alleviates the pressure that drives standard models toward the **outlier-driven rescaling** issue (Qiu et al., 2026).

In this work, we make the following contributions:

1. We formalize the **P0-Sink Circuit**, a two-block subnetwork that exploits the asymmetry of the causal attention mask to generate a high-norm, fixed-direction representation at position zero, providing a consistent reference point for attention heads throughout the network.
2. We propose two parameter-free pre-training methods that promote P0 sink formation from complementary mechanistic perspectives.
3. We conduct comprehensive experiments across diverse learning rate and data scale settings, demonstrating that both methods consistently improve over the Transformer baseline and achieve performance comparable to Gated Attention.

2 How [BOS] Token Affects P0 Sink

Sink Rate Barbero et al. (2025) remove [BOS] and report a weakened sink effect, measured by the **Sink Rate** metric proposed by Gu et al. (2025):

$$\text{Sink}_k^{\epsilon} = \frac{1}{L}\sum_{l=0}^{L-1}\frac{1}{H}\sum_{h=1}^{H}\mathbb{I}(\alpha_k^{l,h} > \epsilon), \tag{1}$$

where $\alpha_k^{l,h}$ denotes the post-softmax attention score assigned to position k at layer l and head h of an L-layer, H-head Transformer, and ϵ is a predefined threshold. However, averaging over all layers obscures layer-wise differences: removing [BOS] may primarily affect early layers while leaving the sink intact elsewhere, a distinction the Sink Rate cannot capture.

Gu et al. (2025) show that the attention sink persists regardless of dataset choice, even on random sequences. However, when the input consists of repeated first tokens, the sink disappears in Mistral and LLaMA models[1]. They attribute this to identical hidden states produced by repeated first tokens

[1]Mistral and LLaMA models use [BOS] as the first token of any sequence, while other LLMs, e.g., Qwen and Pythia, do not.

(a) $\text{Sink}_0^\epsilon[l_{start}:]$ with ablation on `[BOS]` ($\epsilon = 0.3$), following Gu et al. (2025).

Model		$\text{Sink}_0^\epsilon[0:]$	$\text{Sink}_0^\epsilon[1:]$	$\text{Sink}_0^\epsilon[2:]$	$\text{Sink}_0^\epsilon[3:]$
Llama3.1-8B	w/ `[BOS]`	93.69	95.52	95.39	95.23
	w/o `[BOS]`	89.40	92.28	95.35	95.19
Llama3.2-1B	w/ `[BOS]`	90.32	92.07	91.53	90.89
	w/o `[BOS]`	79.49	84.79	90.85	90.15
Llama3.2-3B	w/ `[BOS]`	96.14	96.83	96.71	96.58
	w/o `[BOS]`	89.78	93.11	96.69	96.55
Mistral-7B-v0.3	w/ `[BOS]`	87.45	88.27	87.91	87.50
	w/o `[BOS]`	1.87	1.93	1.92	1.90

(b) Cross-Entropy loss under different first-token repeat counts. **Bold**: smaller loss.

Repeat (n)		n=1	n=2	n=3	n=4	n=5	n=6	n=7	n=8	$\Delta_{1\text{-}8}$
Llama-3.1-8B	w/ `[BOS]`	**2.66**	**2.68**	**2.72**	**2.76**	**2.80**	**2.84**	**2.89**	2.93	0.27
	w/o `[BOS]`	2.79	2.81	2.83	2.85	2.86	2.88	2.89	**2.90**	**0.21**
Llama-3.2-1B	w/ `[BOS]`	**3.22**	**3.27**	**3.36**	**3.46**	3.56	3.68	3.80	3.92	0.70
	w/o `[BOS]`	3.32	3.36	3.42	3.48	**3.53**	**3.58**	**3.62**	**3.66**	**0.34**
Llama-3.2-3B	w/ `[BOS]`	**2.97**	**3.01**	**3.07**	**3.15**	3.23	3.33	3.43	3.53	0.56
	w/o `[BOS]`	3.06	3.10	3.15	3.19	**3.22**	**3.26**	**3.29**	**3.32**	**0.26**
Mistral-7B-v0.3	w/ `[BOS]`	2.44	2.46	2.49	2.54	2.60	2.69	2.79	2.91	0.47
	w/o `[BOS]`	**2.39**	**2.39**	**2.40**	**2.41**	**2.41**	**2.42**	**2.43**	**2.44**	**0.05**

Table 1: Ablation study on `[BOS]` token across different LLMs.

under RoPE (Su et al., 2023). This raises a question: does the P0 sink depend on the learned embedding of `[BOS]`, or on its structural role as the token at position zero?

To answer this question, we examine LLMs that prepend a `[BOS]` token, namely LLaMA and Mistral, which differ in their training regimes. LLaMA uses full-context attention, so `[BOS]` is always visible as the first token throughout training. Mistral uses sliding window attention, causing the first token to fall out of scope after the first slide, making both the `[BOS]` embedding and position 0 invisible to later tokens.

This architectural difference has a functional consequence. Prior work shows that the P0 sink acts as a head-level gating mechanism, where attention is routed to a semantically vacuous position-zero representation to deactivate certain heads (Sandoval-Segura et al., 2026), and that this role can be fulfilled by alternative designs such as gated attention (Qiu et al., 2026; 2025). **Sliding window models like Mistral develop different strategies to fulfill the same functional role.** The same consideration extends to hybrid architectures that interleave sliding window and full attention layers: sliding window layers exhibit no P0 sink while full attention layers do, regardless of whether RoPE is employed (Yang et al., 2025). Xu (2026) also validate that this behavior is universal across architectures. We leave a detailed investigation of these cases for future work.

Modified Sink Rate Since LLMs without a `[BOS]` token also exhibit P0 sinks, there must exist at least one mechanism that gives rise to the P0 sink independently of the `[BOS]` embedding. For LLMs where `[BOS]` remains visible throughout training, the model re-establishes P0 sinks that do not rely on the `[BOS]` embedding within the shallowest layers. To demonstrate this, we introduce a modification to Sink_k^ϵ (Gu et al., 2025):

$$\text{Sink}_k^\epsilon[l_{\text{start}}:] = \frac{1}{L - l_{\text{start}}} \sum_{l=l_{\text{start}}}^{L-1} \frac{1}{H} \sum_{h=1}^{H} \mathbb{I}(\alpha_k^{l,h} > \epsilon), \tag{2}$$

where $l_{\text{start}} \in [0, L)$. This metric shows directly that removing `[BOS]` affects only the shallowest layers, as shown in Table 1a.

This raises a further question: between a P0 representation shaped purely by positional structure and one grounded in the `[BOS]` embedding, which better fulfills the vacuous role that the P0 sink requires? To answer this, we repeat the first token multiple times and measure the resulting change in loss. As shown in Table 1b, experiments are conducted on a subset of the FineWeb-Edu 10B dataset[2]. In the w/ `[BOS]` setting, the repeated token is `[BOS]`; in the w/o `[BOS]` setting, the first content token is repeated. Losses on repeated tokens and the original `[BOS]` position are excluded for fair comparison.

The results in Tables 1a and 1b together support a clear conclusion. For Mistral, `[BOS]` is the sole driver of the sink: removing it collapses the sink rate greatly and directly reduces loss. This is consistent with our earlier discussion: sliding window attention causes `[BOS]` to fall out of scope, so Mistral never develops a positional P0 sink and relies entirely on the `[BOS]` embedding instead. For the LLaMA family, the picture is different. Removing `[BOS]` affects only the first two layers, as shown in Table 1a, where $\text{Sink}_0^\epsilon[2:]$ recovers to near the w/ `[BOS]` level. Correspondingly, repeating a non-`[BOS]` token yields a much smaller difference than repeating `[BOS]` itself, confirming that the positional P0 representation is a more stable semantic vacuum than the `[BOS]` embedding.

[2]Unless specified, we use the first 1024 samples from the first `.parquet` file of FineWeb-Edu 10B dataset as validation set, with a maximum sequence length of 64. These samples are excluded from training.

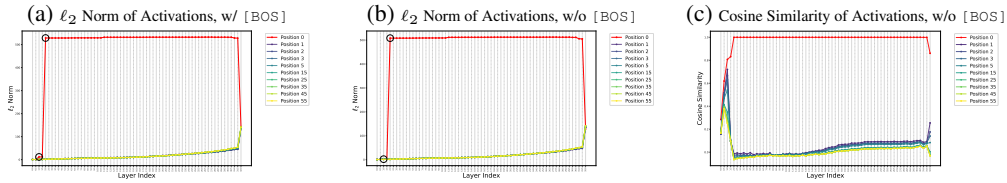


Figure 2: Layer-wise ℓ_2 norm and average cosine similarity of hidden states in LLaMA3.1-8B. Half-integer indices correspond to attention module outputs after the residual connection. Cosine similarity is computed per layer and position as the average cosine similarity between each sample activation and the mean activation across all samples. Both metrics are computed over sequences of up to 64 tokens across 1024 validation samples.

3 How LLMs Sink the Position-Zero Token

3.1 Fixed Representation at Position Zero

Prior work has identified two key observations underlying the P0 sink. Cancedda (2024) show that the hidden state at the [BOS] position develops significantly amplified ℓ_2 norms after passing through certain Transformer layers. Gu et al. (2025) further demonstrate that this norm inflation correlates with disproportionate concentration of attention scores at position zero. Consistent findings are reported by An et al. (2025) and Sun et al. (2024), who observe similar outlier activation patterns from complementary analytical perspectives.

We build on these observations to characterize the mechanism more precisely. As shown in Table 1a and Figure 2, even after removing [BOS], the model localizes the position-zero token and forms a P0 sink within just two Transformer blocks. This is not specific to [BOS]: Li et al. (2026) show that blocking any intermediate token from attending to prior positions is sufficient to induce an attention sink on it, and that artificially amplifying the variance of an arbitrary token's attention output produces the same effect, whereas merely scaling its representation norm does not. The common condition is therefore elevated variance in the attention output, not token identity or representation magnitude.

The MLP sublayer is central to this process. It not only inflates the ℓ_2 norm of the P0 hidden state, but also projects it toward a consistent direction in representation space. Figure 2 illustrates both effects. We quantify the directional convergence via cosine similarity in Figure 2c: values near 1 confirm that, across varying inputs, the P0 representation converges to the same direction after just two Transformer blocks. Since pre-layer RMS normalization (pre-norm) removes magnitude information before each attention and MLP sublayer (Zhang & Sennrich, 2019; Ba et al., 2016), direction is precisely what downstream computation sees, making this convergence functionally significant. We omit the w/ [BOS] condition from Figure 2c, as the repeated presence of the same token embedding at position zero makes directional convergence uninformative as a test of the positional mechanism.

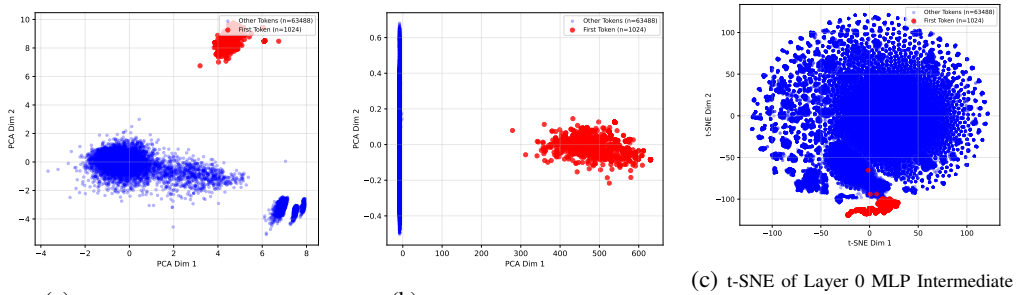
This ℓ_2 amplification serves two functional roles. First, it renders the P0 representation semantically vacuous. Because the MLP output is added to the residual stream, the amplified hidden state at P0 comes to dominate over early layer contributions. Second, it stabilizes the P0 representation across training. Under pre-norm, a high-magnitude vector is less sensitive to gradient updates and more likely to preserve its direction across optimization steps. This stability allows the model to maintain a consistent, distinguishable representation at position zero throughout training, one that attention heads across the network can reliably target.

To formalize the stability of the P0 representation under training, we analyze how pre-norm interacts with gradient updates. The pre-norm transformation applied to a hidden state $\mathbf{x} \in \mathbb{R}^d$ is:

$$\mathrm{Norm}(\mathbf{x}) = \frac{\mathbf{x}}{r(\mathbf{x})}, \quad r(\mathbf{x}) := \sqrt{\frac{1}{d}\mathbf{x}^\top \mathbf{x}}. \tag{3}$$

Its Jacobian with respect to $\mathbf{x}$ is:

$$\mathbf{J}_{\mathrm{Norm}}(\mathbf{x}) = \frac{1}{r(\mathbf{x})}\mathbf{I} - \frac{1}{d}\frac{1}{r^3(\mathbf{x})}\mathbf{x}\mathbf{x}^\top. \tag{4}$$



(a) PCA of Layer 1 MLP Input (b) PCA of Layer 2 Attn Input (c) t-SNE of Layer 0 MLP Intermediate States, w/o Head 29

Figure 3: 2D dimensionality-reduction visualizations of activations in LLaMA3.1-8B with the `[BOS]` token removed. Each point represents the hidden state of one position, over sequences of up to 64 tokens across 1024 samples from validation dataset.

Consider a perturbation $\delta\mathbf{x} \ll \mathbf{x}$ induced by a gradient update during training, decomposed into components parallel and orthogonal to $\mathbf{x}$:

$$\delta\mathbf{x} = \delta\mathbf{x}_{\parallel} + \delta\mathbf{x}_{\perp}, \quad \delta\mathbf{x}_{\parallel} = \alpha\mathbf{x}, \quad \delta\mathbf{x}_{\perp} \cdot \mathbf{x} = 0. \tag{5}$$

The resulting change in the normalized vector is approximately:

$$\mathrm{Norm}(\mathbf{x} + \delta\mathbf{x}) - \mathrm{Norm}(\mathbf{x}) \approx \mathbf{J}_{\mathrm{Norm}}(\mathbf{x})\,\delta\mathbf{x} = \frac{\delta\mathbf{x}_{\perp}}{r(\mathbf{x})}. \tag{6}$$

The parallel component vanishes entirely, since rescaling $\mathbf{x}$ does not change its normalized direction. The orthogonal component, which does alter the direction, is attenuated by $1/r(\mathbf{x})$. Under the standard i.i.d. assumption on training data, gradient noise can be treated as approximately independent of the input (Martens & Grosse, 2020), so $\delta\mathbf{x}_{\perp}$ represents generic directional perturbation. A larger $r(\mathbf{x})$ therefore directly reduces the sensitivity of the normalized direction to gradient updates, providing a formal basis for the directional stability of high-norm representations observed at position zero.

3.2 Position Zero Identification Circuit

We now turn to how the model establishes a positional signature at position zero prior to MLP amplification. Empirically, the position-zero representation is already pushed toward a relatively stable direction across inputs after layer 0, before any strong MLP amplification takes place. Li et al. (2026) identify value aggregation within the attention module as the key cause, but do not provide a mechanistic account of why and how the model exploits this positional asymmetry. Since activations pass through linear layers before any nonlinearity within a Transformer block, PCA provides a natural tool for capturing this linear structure. Figure 3 confirms that positional identification is established before the layer-1 MLP, that is, before the model's second Transformer block. Once certain confounding signals are removed, for example the output of head 29 in layer 0 of LLaMA3.1-8B, the up-projection of the layer-0 MLP already clusters the position-zero states. This subtraction is exact: since attention head outputs are summed in the residual stream, individual head contributions can be isolated and removed precisely. t-SNE offers a more detailed view of this structure than PCA, as it models pairwise distances between activations (Cai & Ma, 2022).

The mechanism underlying this asymmetry is the causal attention mask. Under causal masking, all positions other than zero aggregate diverse context vectors, which reduces the consistency of any shared directional component in their attention outputs. Position zero, by contrast, attends only to itself, so its attention output remains unmixed and preserves its direction more reliably across inputs. This clean signal is what subsequent MLP sublayers can detect and amplify, ultimately producing the stable P0 representation and the corresponding sink pattern.

Motivated by the observation that attention heads' weighted-averaging behaviors play a key role in forming the P0 sink, we develop a simplified theoretical model to estimate the norm of the attention output. We assume that input vectors $\mathbf{x}_0, \ldots, \mathbf{x}_{l-1}$ have unit norm due to pre-layer normalization, i.e., $\|\mathbf{x}_i\|_2 = 1$ for all i, and model attention as a weighted sum over value vectors $\mathbf{v}_0, \ldots, \mathbf{v}_{l-1}$ with weights $\mathbf{p} = (p_0, \ldots, p_{l-1})$.

Cone-based model of value vectors. As shown in Figure 2c, hidden states in LLMs exhibit clear directional bias and are far from isotropic. We capture this by modeling the value vectors $\mathbf{v}_i$ as lying on a fixed-angle cone centered around a unit vector $\mathbf{u} \in \mathbb{R}^d$, each with unit norm and constant cosine similarity α with the cone axis. In practice, the per-channel learnable scaling in RMSNorm can push the realized norm away from 1, so this is an idealization rather than an exact constraint. A convenient construction is:

$$\mathbf{v}_i = \alpha \mathbf{u} + \sqrt{1-\alpha^2}\,\mathbf{s}_i, \tag{7}$$

where each $\mathbf{s}_i$ is a unit vector orthogonal to $\mathbf{u}$, sampled uniformly from the $(d-1)$-dimensional unit sphere and independently across i. Under this construction:

$$\mathbb{E}[\mathbf{v}_i^\top \mathbf{v}_j] = \begin{cases} 1, & i = j, \\ \alpha^2, & i \neq j. \end{cases} \tag{8}$$

Attention output norm. Define the attention output at position $l-1$ as:

$$\mathbf{c} := \sum_{i=0}^{l-1} p_i \mathbf{v}_i. \tag{9}$$

Expanding the squared norm and taking expectation over $\mathbf{v}$ conditioned on $\mathbf{p}$:

$$\mathbb{E}_{\mathbf{v}}[\|\mathbf{c}\|_2^2 \mid \mathbf{p}] = \sum_i p_i^2 \cdot 1 + \sum_{i \neq j} p_i p_j \cdot \alpha^2 = \alpha^2 + (1-\alpha^2) \sum_i p_i^2. \tag{10}$$

Taking expectation over $\mathbf{p}$ then gives:

$$\mathbb{E}[\|\mathbf{c}\|_2^2] = \alpha^2 + (1-\alpha^2)\,\mathbb{E}\left[\sum_i p_i^2\right]. \tag{11}$$

While the full distribution of $\mathbf{p}$ is difficult to characterize analytically, empirical evidence suggests that attention tends to be sparse across positions (Zucchet et al., 2025), implying that $\mathbb{E}\left[\sum_i p_i^2\right]$ decreases monotonically as sequence length l increases. This is consistent with our empirical finding that the ℓ_2 norm of the attention output decreases with sequence length, as shown in Table 2. Position zero is the limiting case: with $p_0 = 1$ enforced by causal masking, $\sum_i p_i^2 = 1$ regardless of sequence length, yielding a strictly higher expected output norm than any other position. This norm asymmetry is the signal that MLP sublayers detect and amplify into the stable P0 representation.

Token Index	0	1	2	4	8	16	32	64
Llama-3.1-8B	0.87	0.82	0.79	0.73	0.66	0.63	0.61	0.59
Llama-3.2-1B	1.04	1.03	0.98	0.96	0.91	0.90	0.88	0.87
Llama-3.2-3B	1.37	1.29	1.21	1.17	1.11	1.10	1.07	1.05

Table 2: ℓ_2 Norm of Layer 0 Attn Out, w/o `[BOS]`

4 Free Lunch for Improving Transformers as LLM Backbones

For years, researchers have sought improved attention mechanisms to strengthen the Transformer decoder as a backbone for auto-regressive LLMs (Peng et al., 2026; Shazeer et al., 2020). Few such efforts have been adopted in modern practice, with the recently notable exceptions of Gated Attention (Qiu et al., 2026). We propose two simple methods that accelerate the formation of a stable P0 sink representation and achieve performance comparable to Gated Attention, while introducing no additional parameters. This makes them particularly attractive during pretraining and pre-filling, the two stages where Gated Attention incurs the most overhead.

4.1 Preliminary

As discussed in Section 2, modern open-source LLMs show no consistent pattern in their handling of the <BOS> token, and no consensus has emerged on whether to include it during pretraining. Our experimental results are consistent with this observation: pretraining under either choice yields little difference in downstream performance, with detailed results reported in Appendix A.

Two additional observations inform our experimental setup. First, training on FineWeb-Edu appears to make WikiText (Merity et al., 2016) an unreliable benchmark: as training progresses, WikiText

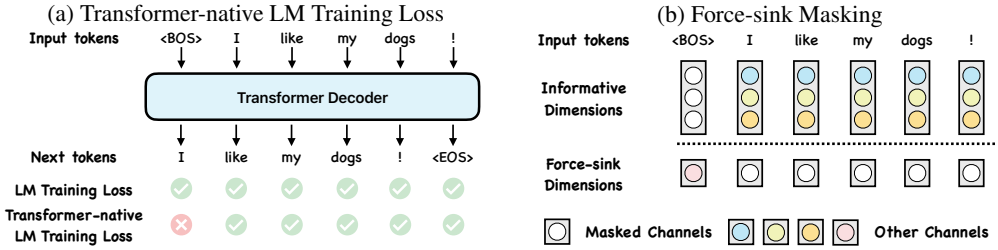


Figure 4: Illustration of the two variants that modify only how the P0 representation is trained, each designed to accelerate the formation process of the P0-Sink Circuit.

perplexity reported by LM-Eval (Gao et al., 2024) degrades, in contrast to the consistent improvement seen on every other benchmark. We therefore omit WikiText results from all tables in the main paper.

Second, under the Chinchilla-optimal setting (Hoffmann et al., 2022), a 1B model trained on 20B tokens has not yet fully formed its sink, as shown in Table 3, consistent with observations in prior work (Xu, 2026). Across our experiments, models trained with <BOS> form the sink faster, whereas models trained without <BOS> eventually form a stronger sink in the shallow layers.

Model		$\text{Sink}_0^\epsilon[0:]$	$\text{Sink}_0^\epsilon[1:]$	$\text{Sink}_0^\epsilon[2:]$	$\text{Sink}_0^\epsilon[4:]$	$\text{Sink}_0^\epsilon[8:]$	$\text{Sink}_0^\epsilon[16:]$
Trained W/ <BOS>	**Baseline-1B**, Chinchilla-optimal	36.33	37.91	39.63	43.53	47.47	49.68
	Baseline-1B, 5× Chinchilla-optimal	76.58	79.90	83.53	87.37	89.26	94.01
Trained W/O <BOS>	**Baseline-1B**, Chinchilla-optimal	10.34	10.79	11.28	12.37	13.59	15.98
	Baseline-1B, 5× Chinchilla-optimal	79.89	83.36	87.15	90.93	91.38	93.28

Table 3: Our modified version of Sink_k^ϵ under <BOS> ablation shows that the sink builds up gradually from bottom layers. Under the popular Chinchilla-optimal setting, a 1B model trained on 20B tokens has not yet finished forming the sink. At 5× this scale, 100B tokens following the FLAME setup (Zhang & Yang, 2025), the sink is clearly visible in both the W/ and W/O <BOS> settings.

This difference follows from the distinct learning targets the two choices induce. Causal masking restricts the first token to attend only to its own value vector, making the QK interaction irrelevant there; both models reduce to a pointwise function, but with different targets. With <BOS>, the target is the marginal distribution over the first real token, a single fixed distribution. Without <BOS>, the target is a bigram distribution indexed by whichever token occupies the first position, a family of distributions.

To see why this affects sink formation, consider the gradient at position zero decomposed into two components:

$$\nabla \mathcal{L}_0 = \nabla \mathcal{L}_0^{\text{sink}} + \nabla \mathcal{L}_0^{\text{pred}}, \tag{12}$$

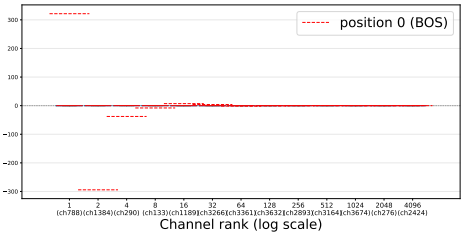
where $\nabla \mathcal{L}_0^{\text{sink}}$ drives the model toward forming the sink circuit and $\nabla \mathcal{L}_0^{\text{pred}}$ drives it toward capturing the prediction target. With <BOS>, the prediction target is a fixed unigram distribution independent of the input, so $\nabla \mathcal{L}_0^{\text{pred}}$ converges faster and exerts diminishing interference on $\nabla \mathcal{L}_0^{\text{sink}}$. Without <BOS>, the prediction target is a bigram distribution conditioned on the first content token, a strictly harder objective that varies across inputs. The corresponding $\nabla \mathcal{L}_0^{\text{pred}}$ therefore remains large throughout training and persistently interferes with $\nabla \mathcal{L}_0^{\text{sink}}$, slowing sink circuit formation. This accounts for the slower sink development we observe in non-<BOS> models under the Chinchilla-optimal setting.

Based on this analysis, although the two choices are nearly indistinguishable in downstream performance, the <BOS> configuration gives every sequence the same fixed starting token, keeping the setup simple and consistent with common practice. We therefore adopt the <BOS>-in-vocabulary setting for all subsequent experiments.

4.2 Transformer-Native LM Training Loss

As established above, the prediction made at the <BOS> position only recovers the marginal distribution over the first real token, a degenerate target that carries no genuine sequence-modeling signal. Moreover, as the gradient decomposition in Section 4.1 shows, the corresponding gradient

(a) Channel values before pre-layer RMS Norm



(b) Channel values after pre-layer RMS Norm

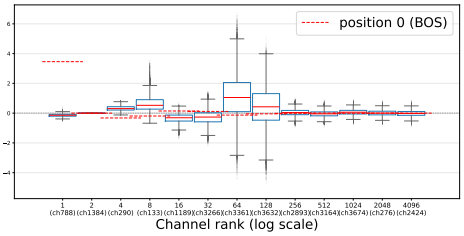


Figure 5: Box plots of hidden-state values across channels in LLaMA3.1-8B at layer 2, with channels ranked by their absolute mean value of `[BOS]`. Since the `[BOS]` hidden state is fixed across samples, it is shown as a red dashed line for each channel. Layers beyond layer 2 exhibit similar patterns. We count over sequences of up to 64 tokens across 1024 samples from validation dataset.

component $\nabla \mathcal{L}_0^{\text{pred}}$ actively interferes with sink circuit formation throughout training. We therefore exclude the cross-entropy term at the <BOS> position from the training objective, as shown in Figure 4a. This simultaneously removes the interfering gradient component and frees the position to act as a dedicated register that holds the sink, motivated by work on register tokens in vision transformers (Darcet et al., 2024; Jiang et al., 2025): a constant, content-free token can absorb the attention sink without degrading the model's representations. We refer to the resulting objective as the **Transformer-Native LM Training Loss**, since it removes a prediction target that does not reflect a genuine sequence-modeling task and aligns the training signal with the sink structure the architecture naturally develops.

4.3 Force-Sink Masking

Returning to Section 3, the P0-Sink Circuit performs two essential functions: identifying the P0 position, and suppressing non-sink channels by amplifying a subset of them to large magnitude. The latter is also a core issue that recent work has sought to address (Qiu et al., 2026), where this behavior is termed **outlier-driven rescaling**.

As shown in Figure 5, approximately 16–32 channels exhibit statistically significant differences in hidden-state values at the `[BOS]` position compared to other positions. After applying the pre-layer RMS Norm with learnable scaling factors, these outlier values are sufficiently suppressed that their statistics become comparable to those of other channels. Notably, however, for the `[BOS]` token, the remaining channels outside this top set are more tightly concentrated around zero relative to other positions.

Based on this observation, we design a single, parameter-free mechanism that realizes both functions simultaneously, as shown in Figure 4b. Let $h_t \in \mathbb{R}^d$ denote the residual-stream hidden state at position t. We fix a binary mask $m \in \{0,1\}^d$ selecting $k \ll d$ channels, and apply it at every layer before pre-norm:

$$h_t \leftarrow \begin{cases} m \odot h_t, & t = 0, \\ (\mathbf{1} - m) \odot h_t, & t \neq 0, \end{cases} \tag{13}$$

where $\odot$ denotes elementwise multiplication and $\mathbf{1} - m$ is the complementary mask over the remaining $d - k$ channels. Applied at every layer, this keeps the P0 representation confined to the same k-dimensional subspace throughout the network, while every other token's representation stays confined to the complementary $(d - k)$-dimensional subspace. No channel is ever active for both P0 and non-P0 tokens at any depth.

This hard partition realizes both functions of the P0-Sink Circuit without any additional parameters. First, it directly achieves identification: since the linear layers, which account for nearly all learnable parameters in the Transformer, receive P0 and non-P0 inputs on disjoint subspaces, each layer naturally learns a distinct sub-mapping for the two cases. Second, it eliminates the pressure toward outlier-driven rescaling: since the $d - k$ non-sink channels are already zeroed at P0, suppression is structural rather than learned, and the model never needs to develop large-magnitude outliers to achieve it.

Method	LR	Sink Rate	Lambada	PIQA	HellaSwag	WinoGrande	ARC_e	ARC_c	Avg.
Baseline*	3×10^{-4}	54.11	38.73	69.10	49.53	53.12	58.16	31.40	52.26
Gated Attention		0.51	**41.78** (+3.05)	70.84 (+1.74)	**51.41** (+1.88)	55.56 (+2.44)	61.32 (+3.16)	33.62 (+2.22)	54.55 (+2.29)
Force-sink Masking		71.27 (+17.16)	41.24 (+2.51)	**70.95** (+1.85)	50.46 (+0.93)	**56.04** (+2.92)	**62.21** (+4.05)	33.36 (+1.96)	**54.60** (+2.34)
TNLM Loss		50.56 (-3.55)	41.72 (+2.99)	70.78 (+1.68)	50.48 (+0.95)	55.01 (+1.89)	59.60 (+1.44)	**33.79** (+2.39)	53.93 (+1.67)
Force-sink Masking+TNLM Loss		**71.77** (+17.66)	41.18 (+2.45)	70.51 (+1.41)	50.83 (+1.30)	54.78 (+1.66)	61.74 (+3.58)	33.02 (+1.62)	54.18 (+1.91)
Baseline*	4×10^{-4}	67.97	40.38	69.97	50.81	54.78	57.62	33.70	53.38
Gated Attention		0.84	42.77 (+2.39)	71.11 (+1.14)	**52.90** (+2.09)	55.09 (+0.31)	62.12 (+4.50)	35.58 (+1.88)	**55.36** (+1.98)
Force-sink Masking		74.68 (+6.71)	**44.71** (+4.33)	71.22 (+1.25)	51.40 (+0.59)	55.01 (+0.23)	**63.38** (+5.76)	**35.75** (+2.05)	55.35 (+1.98)
TNLM Loss		54.92 (-13.05)	42.15 (+1.77)	**71.71** (+1.74)	51.40 (+0.59)	54.30 (-0.48)	58.67 (+1.05)	33.53 (-0.17)	53.92 (+0.55)
Force-sink Masking+TNLM Loss		**74.88** (+6.91)	42.67 (+2.29)	71.27 (+1.30)	51.55 (+0.74)	**56.20** (+1.42)	61.83 (+4.21)	33.70 (+0.00)	54.91 (+1.53)
Baseline*	5×10^{-4}	33.24	39.38	70.51	50.12	55.17	60.90	33.45	54.03
Gated Attention		-	43.78 (+4.40)	71.76 (+1.25)	**53.26** (+3.14)	57.22 (+2.05)	62.12 (+1.22)	**35.49** (+2.04)	55.97 (+1.94)
Force-sink Masking		76.38 (+43.14)	**44.25** (+4.87)	71.38 (+0.87)	52.51 (+2.39)	57.54 (+2.37)	62.88 (+1.98)	35.41 (+1.96)	55.94 (+1.91)
TNLM Loss		57.31 (+24.07)	41.49 (+2.11)	**72.47** (+1.96)	52.42 (+2.30)	57.38 (+2.21)	61.20 (+0.30)	33.70 (+0.25)	55.43 (+1.40)
Force-sink Masking+TNLM Loss		**76.97** (+43.73)	42.48 (+3.10)	71.16 (+0.65)	52.64 (+2.52)	**58.33** (+3.16)	**63.17** (+2.27)	34.73 (+1.28)	**56.01** (+1.98)

Table 4: Ablations under the Chinchilla-optimal setting (Hoffmann et al., 2022), using a 1B model trained on 20B tokens across different learning rates. **Bold** indicates the best performance within the same setting. The star (*) indicates a different model checkpoint from the one used in the previous section. For a fair comparison, we add the same number of parameters to the MLP layer of both the baseline model and our method, since Gated Attention introduces an entire additional linear layer to implement its gating function.

4.4 Experimental Results

Experimental Settings We follow the Chinchilla-optimal setting (Hoffmann et al., 2022) adopted in prior work (Qiu et al., 2025), training on 20B tokens for a 1B-parameter model. All other hyperparameters are aligned with the transformer-1.3B-100B baseline from the FLAME pre-training framework (Zhang & Yang, 2025). Full training details are provided in Appendix B.1.

Table 4 presents an ablation study across three learning rate settings. Both proposed methods consistently improve downstream performance over the baseline. Force-sink Masking outperforms Gated Attention on most tasks, while TNLM Loss alone also yields competitive results. Their combination achieves strong performance across settings. As shown in Appendix C, Force-sink Masking most effectively suppresses outlier activations in the hidden states, reducing peak ℓ_2 from above 2500 to below 20 in the data-saturated setting.

One exception is the drop in Sink Rate for TNLM Loss at 4×10^{-4} (marked in blue). Under this setting, the model does not form a P0 sink until layer 6, resembling the behavior of Qwen3-4B, where the sink emerges in deeper layers with more severely concentrated attention on the P0 token. Further discussion is provided in Appendix D.

More broadly, the results suggest that earlier P0 sink formation is beneficial during pre-training: across comparable settings, a higher Sink Rate consistently corresponds to better downstream performance. Beyond promoting earlier sink formation, both methods also improve task performance directly, supporting the view that a stable P0 sink representation helps stabilize the QK optimization landscape during pre-training. We additionally evaluate both methods under data-saturated settings at $5\times$ the Chinchilla-optimal data scale; results are reported in Appendix E, where Gated Attention shows diminished gains while our methods remain effective.

5 Conclusion

We proposed the **P0-Sink Circuit**, a two-block subnetwork that explains the consistent emergence of the position-zero attention sink in causal LLMs. Position zero attends only to itself under causal masking, producing a directionally stable output that MLP sublayers amplify into a high-norm, fixed-direction representation, without relying on any semantic content.

We further proposed **TNLM Loss**, which removes the cross-entropy term at `[BOS]` to free it as a dedicated attention register, and **Force-sink Masking**, which hard-partitions hidden dimensions between position zero and all other positions, structurally realizing the P0-Sink Circuit and alleviating outlier-driven rescaling. Both are parameter-free and require no architectural changes.

Experiments across diverse learning rate and data scale settings confirm that earlier P0 sink formation correlates with better downstream performance, both methods consistently improve over the Transformer baseline, and Force-sink Masking achieves performance comparable to Gated Attention.

AI USE STATEMENT

We used generative AI tools to assist with writing clarity and code debugging. All AI-assisted content has been reviewed by the authors, and we take full responsibility for the final text, claims, and artifacts in this work.

ETHICS STATEMENT

This work raises a fairness concern regarding the Chinchilla-optimal experimental setting: Transformer models require forming the P0 sink to carry out certain functions, yet under Chinchilla-optimal training budgets this process is incomplete, potentially invalidating comparisons made under this setting. Our two proposed methods address this directly by inducing a stable P0 sink even within Chinchilla-optimal budgets, offering a practical alternative that reduces the computational cost of future experiments.

REPRODUCIBILITY STATEMENT

All experiments use open-source datasets and training repositories. We provide final model checkpoints for researchers with limited compute access. Pre-training experiments can be reproduced on 8 GPUs with at least 100 GiB of per-GPU memory using FSDP.

REFERENCES

Yongqi An, Xu Zhao, Tao Yu, Ming Tang, and Jinqiao Wang. Systematic outliers in large language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=rLX7Vyyzus.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization, 2016. URL https://arxiv.org/abs/1607.06450.

Federico Barbero, Álvaro Arroyo, Xiangming Gu, Christos Perivolaropoulos, Michael Bronstein, Petar Veličković, and Razvan Pascanu. Why do llms attend to the first token?, 2025. URL https://arxiv.org/abs/2504.02732.

T. Tony Cai and Rong Ma. Theoretical foundations of t-sne for visualizing high-dimensional clustered data, 2022. URL https://arxiv.org/abs/2105.07536.

Nicola Cancedda. Spectral filters, dark signals, and attention sinks, 2024. URL https://arxiv.org/abs/2402.09221.

Zhuoming Chen, Ranajoy Sadhukhan, Zihao Ye, Yang Zhou, Jianyu Zhang, Niklas Nolte, Yuandong Tian, Matthijs Douze, Leon Bottou, Zhihao Jia, and Beidi Chen. Magicpig: Lsh sampling for efficient llm generation, 2024. URL https://arxiv.org/abs/2410.16179.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers, 2024. URL https://arxiv.org/abs/2309.16588.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. https://transformer-circuits.pub/2021/framework/index.html.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. The language model evaluation harness, 07 2024. URL https://zenodo.org/records/12608602.

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, 2025. URL https://arxiv.org/abs/2410.10781.

Chi Han, Qifan Wang, Hao Peng, Wenhan Xiong, Yu Chen, Heng Ji, and Sinong Wang. LM-infinite: Zero-shot extreme length generalization for large language models. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 3991–4008, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.222. URL https://aclanthology.org/2024.naacl-long.222/.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models, 2022. URL https://arxiv.org/abs/2203.15556.

Nick Jiang, Amil Dravid, Alexei Efros, and Yossi Gandelsman. Vision transformers don't need trained registers, 2025. URL https://arxiv.org/abs/2506.08010.

Siquan Li, Kaiqi Jiang, Jiacheng Sun, and Tianyang Hu. The structural origin of attention sink: Variance discrepancy, super neurons, and dimension disparity, 2026. URL https://arxiv.org/abs/2605.06611.

James Martens and Roger Grosse. Optimizing neural networks with kronecker-factored approximate curvature, 2020. URL https://arxiv.org/abs/1503.05671.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models, 2016. URL https://arxiv.org/abs/1609.07843.

Runyu Peng, Yunhua Zhou, Demin Song, Kai Lv, Bo Wang, Qipeng Guo, and Xipeng Qiu. Explicit multi-head attention for inter-head interaction in large language models, 2026. URL https://arxiv.org/abs/2601.19611.

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, 2025. URL https://arxiv.org/abs/2505.06708.

Zihan Qiu, Zeyu Huang, Kaiyue Wen, Peng Jin, Bo Zheng, Yuxin Zhou, Haofeng Huang, Zekun Wang, Xiao Li, Huaqing Zhang, Yang Xu, Haoran Lian, Siqi Zhang, Rui Men, Jianwei Zhang, Ivan Titov, Dayiheng Liu, Jingren Zhou, and Junyang Lin. A unified view of attention and residual sinks: Outlier-driven rescaling is essential for transformer training, 2026. URL https://arxiv.org/abs/2601.22966.

Pedro Sandoval-Segura, Xijun Wang, Ashwinee Panda, Micah Goldblum, Ronen Basri, Tom Goldstein, and David Jacobs. Identifying and evaluating inactive heads in pretrained llms, 2026. URL https://arxiv.org/abs/2504.03889.

Noam Shazeer, Zhenzhong Lan, Youlong Cheng, Nan Ding, and Le Hou. Talking-heads attention, 2020. URL https://arxiv.org/abs/2003.02436.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding, 2023. URL https://arxiv.org/abs/2104.09864.

Mingjie Sun, Xinlei Chen, J. Zico Kolter, and Zhuang Liu. Massive activations in large language models, 2024. URL https://arxiv.org/abs/2402.17762.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks, 2024. URL https://arxiv.org/abs/2309.17453.

Yongzhong Xu. When do attention circuits form? developmental trajectories of capability and attention-sink emergence across three 1b-classarchitectures, 2026. URL https://arxiv.org/abs/2606.02378.

Bowen Yang, Bharat Venkitesh, Dwarak Talupuru, Hangyu Lin, David Cairuz, Phil Blunsom, and Acyr Locatelli. Rope to nope and back again: A new hybrid attention strategy, 2025. URL https://arxiv.org/abs/2501.18795.

Zhongzhi Yu, Zheng Wang, Yonggan Fu, Huihong Shi, Khalid Shaikh, and Yingyan Celine Lin. Unveiling and harnessing hidden attention sinks: Enhancing large language models without training through attention calibration, 2024. URL https://arxiv.org/abs/2406.15765.

Biao Zhang and Rico Sennrich. Root mean square layer normalization, 2019. URL https://arxiv.org/abs/1910.07467.

Stephen Zhang, Mustafa Khan, and Vardan Papyan. Attention sinks: A 'catch, tag, release' mechanism for embeddings, 2025. URL https://arxiv.org/abs/2502.00919.

Yu Zhang and Songlin Yang. Flame: Flash language modeling made easy, January 2025. URL https://github.com/fla-org/flame.

Nicolas Zucchet, Francesco D'Angelo, Andrew Kyle Lampinen, and Stephanie C.Y. Chan. The emergence of sparse attention: impact of data distribution and benefits of repetition. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=jMhRbV47pS.

A Ablation Study on the [BOS] Token

Model		Wikitext bits per byte↓	Wikitext byte ppl↓	Lambada acc↑	PIQA acc↑	HellaSwag acc↑	WinoGrande acc↑	ARC_e acc↑	ARC_c acc↑
Trained W/ <BOS>	**Baseline**, Chinchilla-optimal	1.9132	3.7664	39.96	70.57	49.70	54.62	59.22	33.19
	Baseline, 5x Chinchilla-optimal	2.2953	4.9084	47.80	74.16	59.13	59.43	68.69	39.16
Trained W/O <BOS>	**Baseline**, Chinchilla-optimal	1.8124	3.5122	39.57	71.44	50.11	52.09	61.32	34.04
	Baseline, 5x Chinchilla-optimal	2.3440	5.0772	47.88	74.10	59.03	59.27	68.56	40.27

Table 5: Downstream Performance for the [BOS] ablation.

As discussed in Section 2, modern open-source LLMs show no consistent pattern in their handling of the [BOS] token, and no consensus has emerged on whether to include it during pre-training. Our experimental results are consistent with this observation: pre-training under either choice yields little difference in downstream performance, as shown in Table 5.

We also note that training on FineWeb-Edu makes WikiText (Merity et al., 2016) an unreliable benchmark. As training progresses, WikiText perplexity reported by LM-Eval (Gao et al., 2024) degrades, in contrast to the consistent improvement observed on every other benchmark. We therefore omit WikiText results from all tables in the main paper.

B Experimental Details

B.1 Experimental Setups

	Baseline for [BOS] Ablation	Gated Attention	Baseline for Main Exp.
Model architecture			
Parameters	≈1.36B (≈1.23B non-emb)	≈1.46B (≈1.33B non-emb)	≈1.44B (≈1.31B non-emb)
Hidden size d_{model}		2048	
Layers		24	
Attention heads		32	
KV heads		32	
Attention linear layers	4 (WQ, WK, WV, WO)	5 (+ output gate W_θ)	4 (WQ, WK, WV, WO)
FFN activation		SwiGLU	
FFN intermediate dim	5632	5632	6144
FFN ratio	8	8	9
Normalization		RMSNorm, $\epsilon = 10^{-6}$	
Positional encoding		RoPE, $\theta = 10^4$	
Context length		2048	
Vocabulary		32,000	
Tied embeddings		No	
Init. std		0.02	
Training / optimization			
Global batch		256 seqs × 2048 = 0.5M tokens/step	
Sequence length		2048	
Optimizer		AdamW (β_1=0.9, β_2=0.95, ϵ=10^{-15})	
Weight decay		0.1	
Peak learning rate		3×10^{-4}	
LR schedule		cosine decay to $0.1\times$ peak learning rate	
Warmup		1024 steps (≈0.54B tokens)	
Gradient clipping		1.0 (global norm)	
Precision		bfloat16 mixed precision (fp32 reduce)	
Data		FineWeb-Edu (streaming, online tokenization)	
Seed		42	

Table 6: Model and training configuration for the three 1B runs. All hyperparameters, data, and seed are shared; the models differ only in the attention output gate (Gated Attention) and the FFN ratio. The FFN ratio is defined relative to a single attention linear layer in Flame's documentation; note that the Flame repository incorrectly reports half this value for LLaMA-style FFN, and our configurations reflect the corrected definition.

We run experiments at peak learning rates of 3×10^{-4}, 4×10^{-4}, and 5×10^{-4} under the Chinchilla-optimal setting for our main experiments, training for 40,960 steps (∼20B token). For the $5\times$

Chinchilla-optimal (data-saturated) setting in Appendix E, we train for 204,800 steps ($\sim$100B token). For a fair parameter count comparison, we allocate the additional parameters to the FFN rather than introducing an attention output gate as in Gated Attention; **this also reduces the number of additional activations introduced during training.**

Experiment-Specific Settings For **TNLM Loss**, no additional configuration is required beyond excluding the `[BOS]` position from the cross-entropy loss. For **Force-sink Masking**, we use $k = 16$ channels as the Force-sink dimensions (Figure 4b). In the data-saturated experiments reported in Appendix E, Force-sink Masking$^+$ uses $k = 32$. As discussed in Section 4.3, there is no principled criterion for choosing k; we adopt $k = 16$ for the Chinchilla-optimal setting as it performs well, and extend to $k = 32$ for the data-saturated setting where $k = 16$ yields suboptimal results. In practice, we reduce the model's hidden dimensions by k via masking rather than appending extra dimensions, ensuring a fair parameter comparison with the baseline. The baseline model is shared across other experiments and thus it does not use an output normalization layer right before lm head layer; we find no evidence that this affects the validity of our comparisons. Following Yang et al. (2025), we do not apply QK-norm to any of our models.

B.2 Experimental Visualization

As noted above, Gated Attention introduces an entire additional linear layer in the attention module, resulting in increased activation memory overhead during pre-training, as shown in Figure 6.

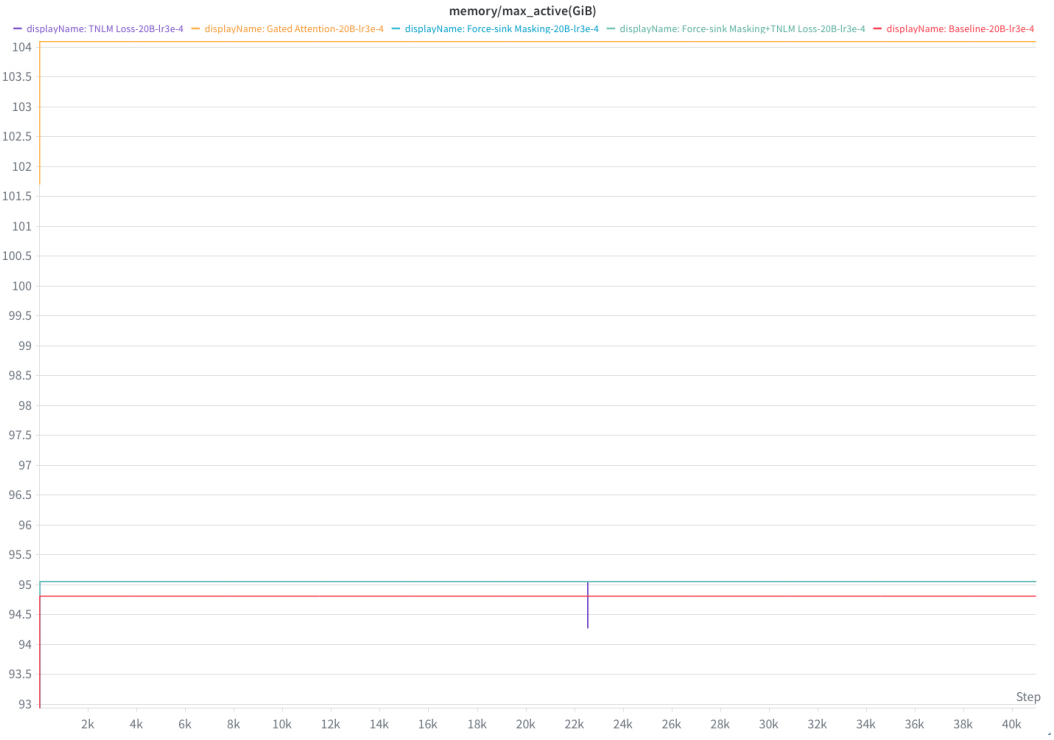


Figure 6: Memory occupation exported from Weights & Biases. A single representative run is shown; all other configurations exhibit identical behavior.

Despite this overhead, our methods achieve comparable loss to Gated Attention during pre-training.

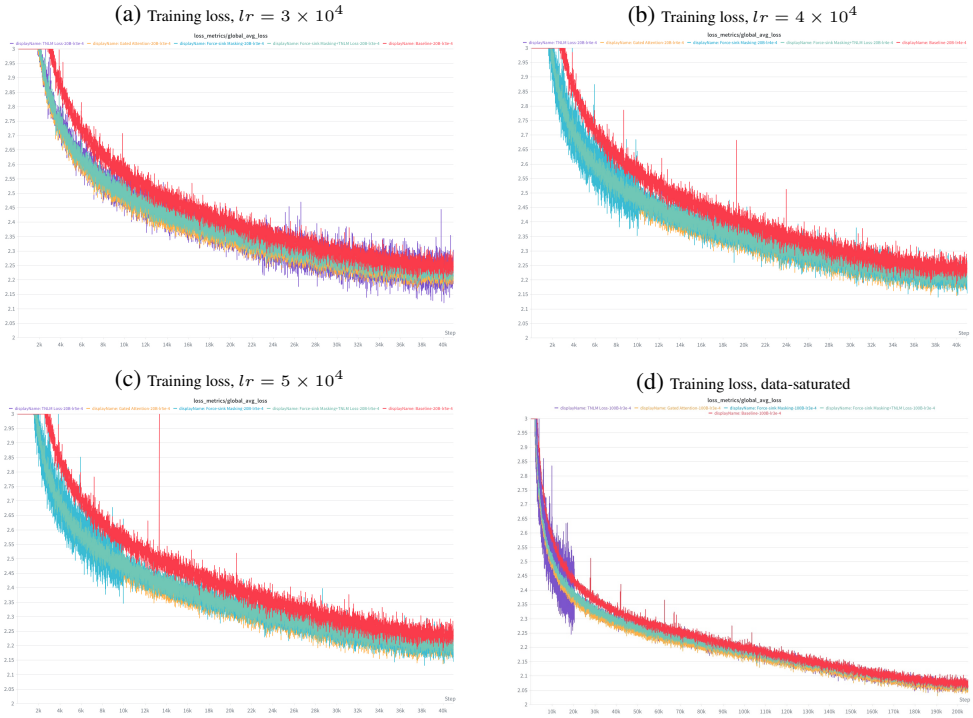


Figure 7: Training loss of all the models in different training settings.

B.3 Code and Weights

The code is at https://anonymous.4open.science/r/flash-linear-attention-6220/, and the checkpoints are at https://huggingface.co/Anonymous2027.

C Outliers Values in Activations

As discussed in Section 3.1, the co-occurrence of attention sinks and outlier activation values is well documented (An et al., 2025; Gu et al., 2025; Jiang et al., 2025; Sun et al., 2024), and our theory provides a mechanistic account of the **outlier-driven rescaling** phenomenon (Qiu et al., 2026). However, as we show in this section, outlier values are not a necessary condition for the attention sink: the two phenomena can be decoupled.

As shown in Table 4, Force-sink Masking yields a higher Sink Rate than the baseline, indicating a stronger P0 sink. Yet the corresponding hidden-state ℓ_2 norms are substantially smaller, demonstrating that a strong attention sink can emerge without large activation outliers.

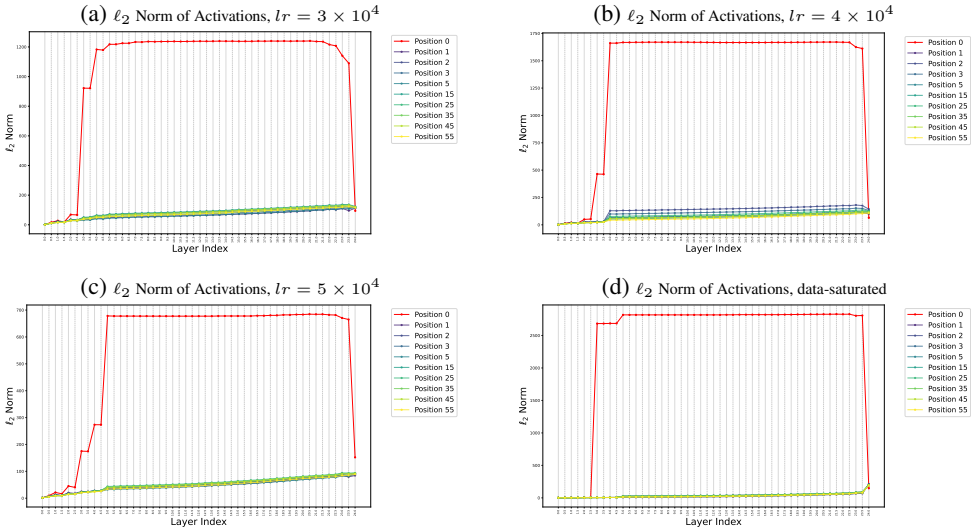


Figure 8: Layer-wise ℓ_2 norm of hidden states in baseline model. Half-integer indices correspond to attention module outputs after the residual connection. Metrics are computed over sequences of up to 64 tokens across 1024 validation samples.

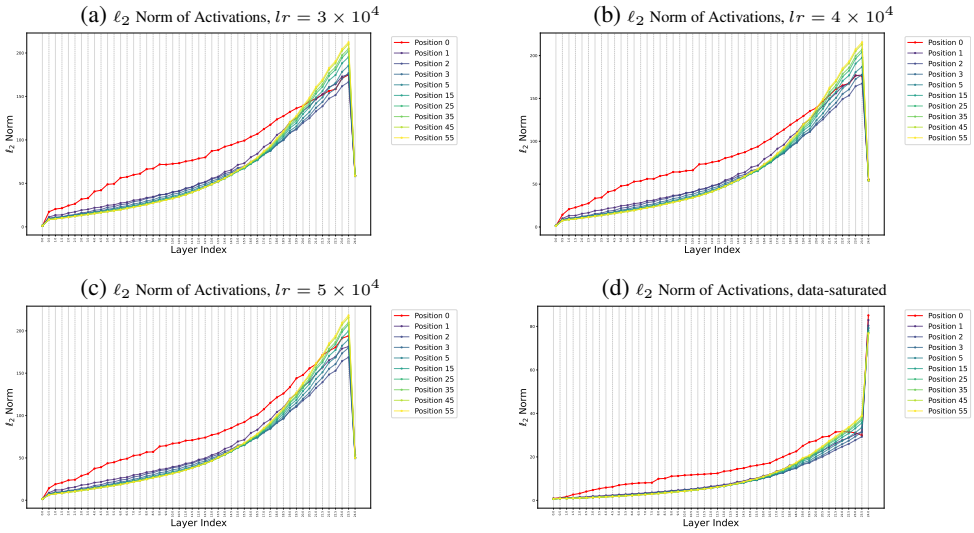


Figure 9: Layer-wise ℓ_2 norm of hidden states in gated attention model. Half-integer indices correspond to attention module outputs after the residual connection. Metrics are computed over sequences of up to 64 tokens across 1024 validation samples.

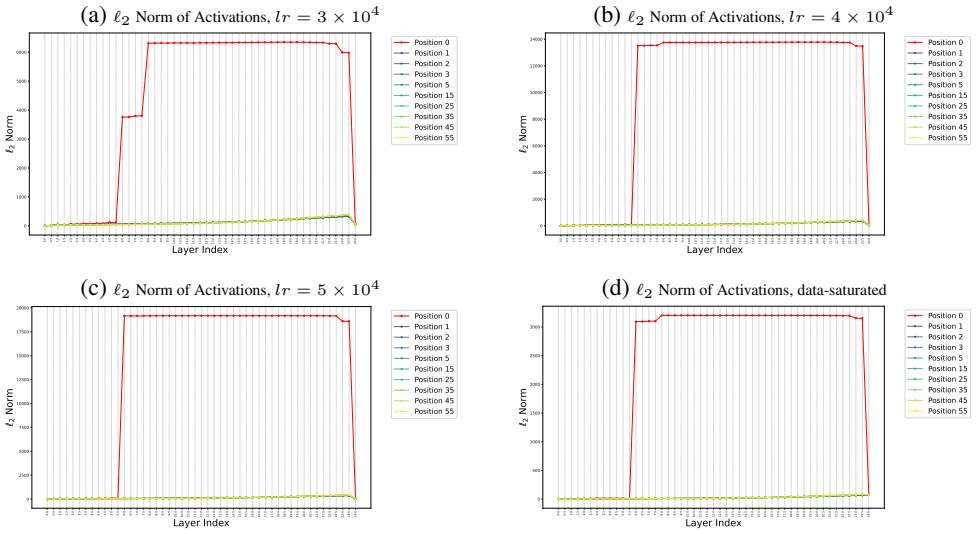


Figure 10: Layer-wise ℓ_2 norm of hidden states in transformer-native lm loss model. Half-integer indices correspond to attention module outputs after the residual connection. Metrics are computed over sequences of up to 64 tokens across 1024 validation samples.

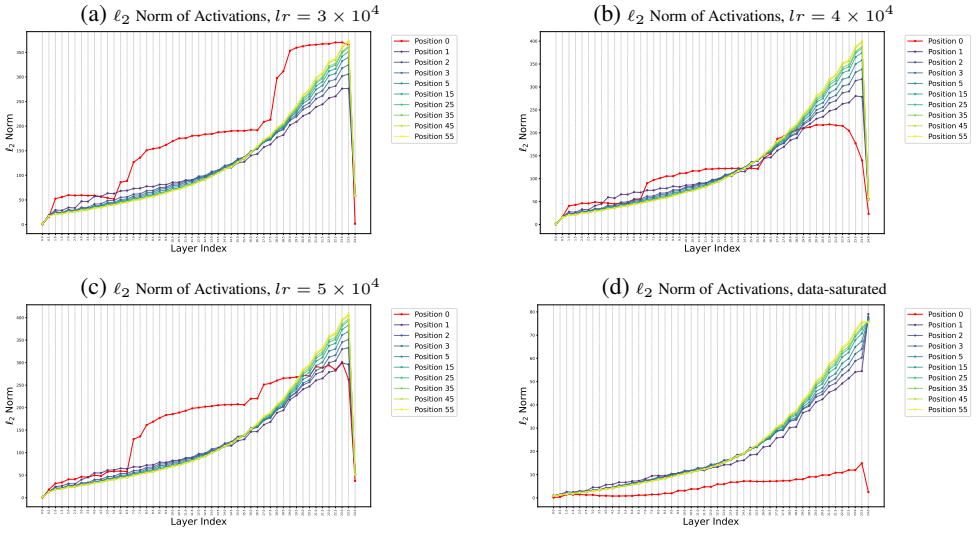


Figure 11: Layer-wise ℓ_2 norm of hidden states in force-sink masking model. Half-integer indices correspond to attention module outputs after the residual connection. Metrics are computed over sequences of up to 64 tokens across 1024 validation samples.

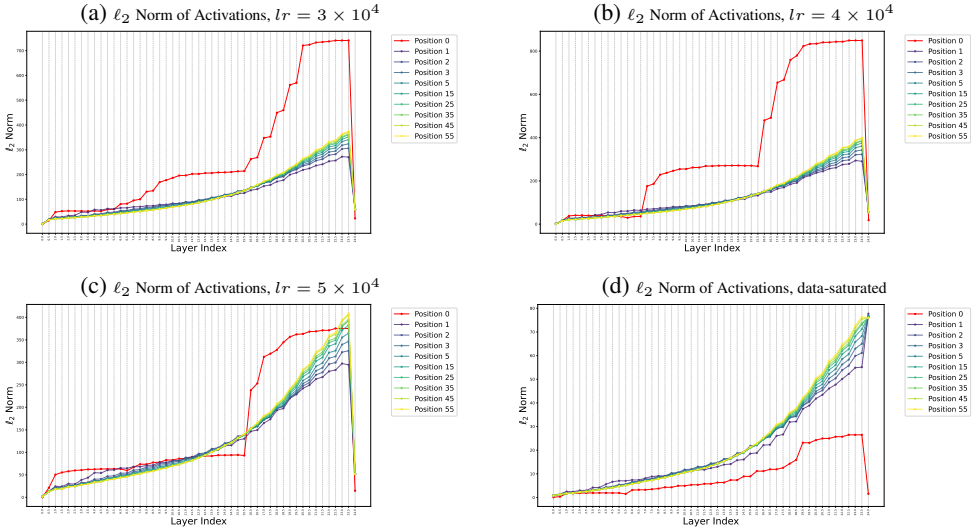


Figure 12: Layer-wise ℓ_2 norm of hidden states in force-sink masking + transformer-native lm loss model. Half-integer indices correspond to attention module outputs after the residual connection. Metrics are computed over sequences of up to 64 tokens across 1024 validation samples.

D SINK RATE ANALYSIS FOR MAIN EXPERIMENT

As shown in Tables 7 and 8, the TNLM Loss model does not form a P0 sink until layer 6[3], where it emerges with more severely concentrated attention on the P0 token. This behavior closely resembles the sink pattern of Qwen3-4B, as shown in Table 9 and Figure 13. For convenience, we reproduce Figure 10 as Figure 14.

Method	**LR**	$\text{Sink}_0^\epsilon[0:]$	$\text{Sink}_0^\epsilon[1:]$	$\text{Sink}_0^\epsilon[2:]$	$\text{Sink}_0^\epsilon[4:]$	$\text{Sink}_0^\epsilon[8:]$	$\text{Sink}_0^\epsilon[16:]$
Baseline*	3×10^{-4}	54.11	56.46	59.03	62.06	70.99	71.59
Gated Attention		0.51	0.53	0.51	0.56	0.63	0.68
Force-sink Masking		71.77	74.89	74.60	74.59	74.28	76.42
TNLM Loss		50.56	52.76	55.15	60.67	69.74	71.86
Force-sink Masking+TNLM Loss		71.27	74.36	74.05	74.06	73.84	75.71
Baseline*	4×10^{-4}	67.97	70.92	74.15	78.32	81.57	84.19
Gated Attention		0.85	0.88	0.86	0.93	1.05	0.96
Force-sink Masking		74.88	78.14	77.77	77.56	77.34	79.34
TNLM Loss		54.92	57.31	59.92	65.91	75.14	77.86
Force-sink Masking+TNLM Loss		74.68	77.93	77.49	77.42	77.69	80.13
Baseline*	5×10^{-4}	33.24	34.68	36.24	39.69	42.76	49.28
Gated Attention		0.97	1.01	1.01	1.09	1.27	1.59
Force-sink Masking		76.97	80.32	80.03	80.12	79.90	81.42
TNLM Loss		57.31	59.80	62.52	68.78	77.09	79.69
Force-sink Masking+TNLM Loss		76.38	79.70	79.35	79.52	80.26	82.05

Table 7: Sink scores at $\epsilon = 0.3$ for the 3 learning rates under Chinchilla-optimal setting.

Method	**LR**	$\text{Sink}_0^\epsilon[0:]$	$\text{Sink}_0^\epsilon[1:]$	$\text{Sink}_0^\epsilon[2:]$	$\text{Sink}_0^\epsilon[4:]$	$\text{Sink}_0^\epsilon[8:]$	$\text{Sink}_0^\epsilon[16:]$
Baseline	3×10^{-4}	22.73	23.72	24.80	26.11	31.06	33.60
Gated Attention		0.13	0.13	0.12	0.13	0.14	0.12
Force-sink Masking		61.04	63.69	62.89	62.34	60.89	63.63
TNLM Loss		38.75	40.43	42.27	46.50	54.39	58.60
Force-sink Masking+TNLM Loss		61.67	64.35	63.59	63.31	61.05	64.19
Baseline	4×10^{-4}	42.31	44.15	46.16	48.56	51.60	56.03
Gated Attention		0.30	0.31	0.28	0.31	0.33	0.22
Force-sink Masking		65.87	68.73	67.88	67.31	66.62	69.39
TNLM Loss		44.50	46.44	48.55	53.40	61.34	66.39
Force-sink Masking+TNLM Loss		65.58	68.43	67.71	66.94	65.71	68.08
Baseline	5×10^{-4}	11.67	12.18	12.73	13.99	15.49	18.70
Gated Attention		0.30	0.31	0.30	0.33	0.38	0.36
Force-sink Masking		68.53	71.51	70.82	70.38	70.16	72.44
TNLM Loss		47.90	49.99	52.26	57.49	64.41	68.42
Force-sink Masking+TNLM Loss		69.15	72.16	71.58	71.36	70.26	72.45

Table 8: Sink scores at $\epsilon = 0.4$ for the 3 learning rates under Chinchilla-optimal setting.

Model	ϵ	$\text{Sink}_0^\epsilon[0:]$	$\text{Sink}_0^\epsilon[1:]$	$\text{Sink}_0^\epsilon[2:]$	$\text{Sink}_0^\epsilon[4:]$	$\text{Sink}_0^\epsilon[8:]$	$\text{Sink}_0^\epsilon[16:]$
Qwen3-4B	0.3	68.47	70.43	72.50	76.92	84.54	84.66
Qwen3-4B	0.4	59.51	61.21	63.01	66.92	73.37	75.40

Table 9: Sink scores for Qwen3-4B at $\epsilon \in \{0.3, 0.4\}$.

[3]For brevity, only two exponent settings are presented in the table.

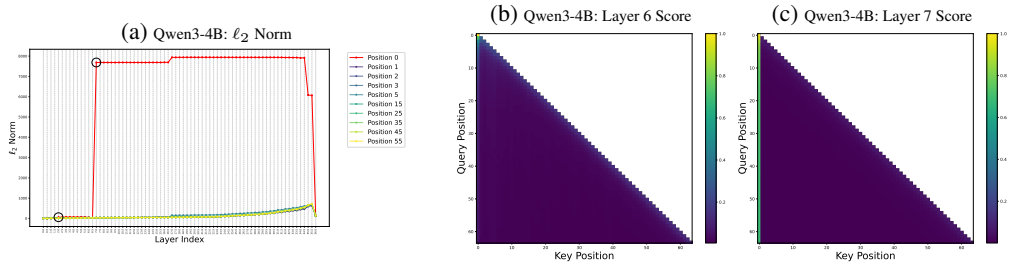


Figure 13: Layer-wise ℓ_2 norm and attention sink visualization for Qwen3-4B. The attention sink becomes prominent after layer 7, coinciding with the sharp increase in the ℓ_2 norm at position zero (P0).

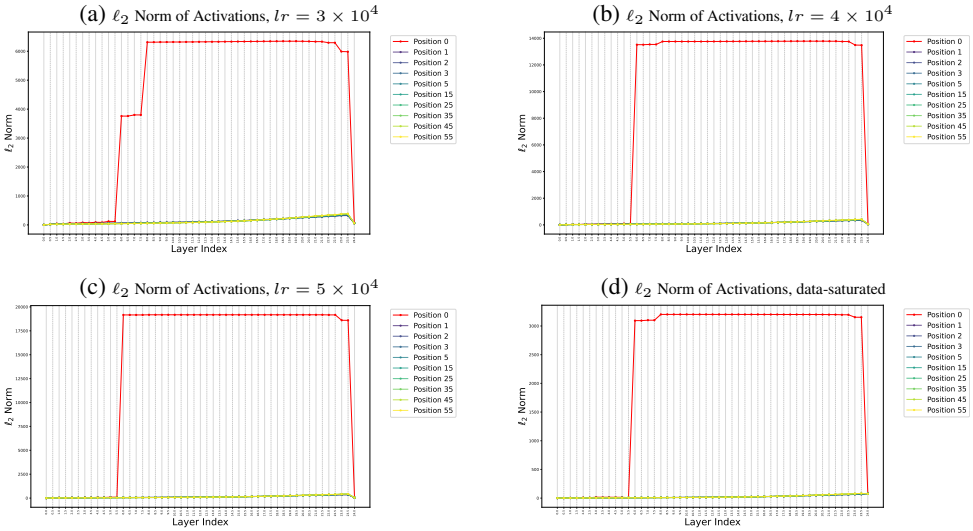


Figure 14: Layer-wise ℓ_2 norm of hidden states in transformer-native lm loss model. Half-integer indices correspond to attention module outputs after the residual connection. Metrics are computed over sequences of up to 64 tokens across 1024 validation samples.

E Data-saturated Experiment

As shown in Tables 5 and 3, baseline models in the Chinchilla-optimal setting have not yet completed P0 sink formation, making comparisons against methods that accelerate this process somewhat unfair, as the baseline still suffers from instability in the attention QK inner-product space. We therefore run data-saturated experiments at $5\times$ the Chinchilla-optimal budget. Under this setting, Gated Attention shows no advantage over the standard Transformer with an equivalent number of parameters added to the MLP. In contrast, TNLM Loss and Force-sink Masking$^+$ ($k = 32$ Force-sink dimensions, Figure 4b) consistently outperform the baseline with fewer FLOPs.

Step	Method	Sink Rate	Lambada	PIQA	HellaSwag	WinoGrande	ARC_e	ARC_c	Avg.
200704	**Baseline**	85.53	49.25	74.21	60.10	59.27	**74.03**	**41.89**	61.90
	Gated Attention	2.33	51.31 (+2.06)	**75.03** (+0.82)	**60.69** (+0.59)	60.93 (+1.66)	71.25 (-2.78)	40.36 (-1.53)	61.65 (-0.25)
	Force-sink Masking	92.65	51.56 (+2.31)	73.94 (-0.27)	60.11 (+0.01)	59.75 (+0.48)	73.11 (-0.92)	40.02 (-1.87)	61.39 (-0.51)
	Force-sink Masking$^+$	90.04	50.73 (+1.48)	74.21 (+0.00)	59.94 (-0.16)	**61.01** (+1.74)	73.61 (-0.42)	40.78 (-1.11)	61.91 (+0.01)
	TNLM Loss	72.27	51.10 (+1.85)	74.86 (+0.65)	60.26 (+0.16)	60.30 (+1.03)	73.53 (-0.50)	41.72 (-0.17)	**62.13** (+0.23)
	Force-sink Masking+TNLM Loss	92.97	51.14 (+1.89)	73.45 (-0.76)	60.34 (+0.24)	59.12 (-0.15)	72.10 (-1.93)	39.51 (-2.38)	60.90 (-1.00)
	Force-sink Masking$^+$+TNLM Loss	92.55	**52.55** (+3.30)	74.81 (+0.60)	60.57 (+0.47)	58.96 (-0.31)	73.36 (-0.67)	40.27 (-1.62)	61.59 (-0.31)
202752	**Baseline**	85.61	48.11	74.10	59.70	59.12	**73.36**	**40.87**	61.43
	Gated Attention	2.44	**51.43** (+3.32)	**75.52** (+1.42)	**60.74** (+1.04)	**60.69** (+1.57)	71.72 (-1.64)	39.42 (-1.45)	**61.62** (+0.19)
	Force-sink Masking	92.65	49.23 (+1.12)	73.56 (-0.54)	60.26 (+0.56)	60.22 (+1.10)	73.11 (-0.25)	39.85 (-1.02)	61.40 (-0.03)
	Force-sink Masking$^+$	90.16	50.48 (+2.37)	74.16 (+0.06)	60.23 (+0.53)	59.19 (+0.07)	72.43 (-0.93)	40.78 (-0.09)	61.36 (-0.07)
	TNLM Loss	72.30	50.94 (+2.83)	74.37 (+0.27)	60.15 (+0.45)	60.30 (+1.18)	73.02 (-0.34)	40.27 (-0.60)	**61.62** (+0.19)
	Force-sink Masking+TNLM Loss	92.99	49.91 (+1.80)	73.12 (-0.98)	60.27 (+0.57)	60.62 (+1.50)	71.25 (-2.11)	39.59 (-1.28)	60.97 (-0.46)
	Force-sink Masking$^+$+TNLM Loss	92.59	50.94 (+2.83)	74.48 (+0.38)	60.49 (+0.79)	59.19 (+0.07)	73.11 (-0.25)	40.27 (-0.60)	61.51 (+0.08)
204800	**Baseline**	85.57	49.37	73.88	59.99	59.75	**73.48**	40.78	61.58
	Gated Attention	2.37	**51.64** (+2.27)	**74.97** (+1.09)	**60.74** (+0.75)	**59.83** (+0.08)	72.01 (-1.47)	40.27 (-0.51)	61.56 (-0.01)
	Force-sink Masking	92.65	50.38 (+1.01)	73.56 (-0.32)	60.13 (+0.14)	59.27 (-0.48)	72.47 (-1.01)	40.19 (-0.59)	61.12 (-0.45)
	Force-sink Masking$^+$	90.10	49.74 (+0.37)	74.86 (+0.98)	60.40 (+0.41)	59.12 (-0.63)	73.06 (-0.42)	**42.06** (+1.28)	**61.90** (+0.33)
	TNLM Loss	72.29	51.10 (+1.73)	74.05 (+0.17)	60.46 (+0.47)	59.35 (-0.40)	72.69 (-0.79)	39.33 (-1.45)	61.18 (-0.40)
	Force-sink Masking+TNLM Loss	93.00	50.34 (+0.97)	73.45 (-0.43)	60.48 (+0.49)	**59.83** (+0.08)	71.80 (-1.68)	40.27 (-0.51)	61.17 (-0.41)
	Force-sink Masking$^+$+TNLM Loss	92.59	51.48 (+2.11)	74.43 (+0.55)	60.58 (+0.59)	59.51 (-0.24)	72.64 (-0.84)	40.19 (-0.59)	61.47 (-0.11)

Table 10: Downstream Performance for the last 3 checkpoints under 5× Chinchilla-optimal setting.

Step	Method	$\text{Sink}_0^\epsilon[0:]$	$\text{Sink}_0^\epsilon[1:]$	$\text{Sink}_0^\epsilon[2:]$	$\text{Sink}_0^\epsilon[4:]$	$\text{Sink}_0^\epsilon[8:]$	$\text{Sink}_0^\epsilon[16:]$
200704	Baseline	85.53	89.25	93.31	97.64	97.36	97.72
	Gated Attention	2.33	2.39	2.42	2.66	3.15	4.76
	Force-sink Masking	92.65	96.68	96.53	96.19	95.49	94.97
	Force-sink Masking$^+$	92.35	96.37	96.20	95.92	95.41	94.64
	TNLM Loss	72.27	75.41	78.84	86.72	96.26	94.73
	Force-sink Masking+TNLM Loss	92.97	97.01	96.88	96.56	95.72	95.50
	Force-sink Masking$^+$+TNLM Loss	92.55	96.57	96.42	96.06	95.40	95.05
202752	Baseline	85.61	89.33	93.39	97.73	97.45	97.78
	Gated Attention	2.44	2.50	2.54	2.80	3.32	5.03
	Force-sink Masking	92.65	96.68	96.53	96.18	95.49	94.95
	Force-sink Masking$^+$	92.43	96.45	96.29	96.02	95.52	94.78
	TNLM Loss	72.30	75.45	78.88	86.76	96.30	94.78
	Force-sink Masking+TNLM Loss	92.99	97.03	96.89	96.58	95.75	95.57
	Force-sink Masking$^+$+TNLM Loss	92.59	96.61	96.46	96.10	95.46	95.14
204800	Baseline	85.57	89.29	93.35	97.68	97.42	97.73
	Gated Attention	2.37	2.42	2.46	2.70	3.22	4.78
	Force-sink Masking	92.67	96.70	96.55	96.21	95.53	95.05
	Force-sink Masking$^+$	92.40	96.42	96.25	95.97	95.47	94.70
	TNLM Loss	72.29	75.44	78.86	86.74	96.26	94.72
	Force-sink Masking+TNLM Loss	93.00	97.04	96.90	96.60	95.76	95.60
	Force-sink Masking$^+$+TNLM Loss	92.59	96.62	96.47	96.11	95.44	95.11

Table 11: Sink scores at $\epsilon = 0.3$ for the last 3 checkpoints under 5× Chinchilla-optimal setting.

Step	Method	$\text{Sink}_0^\epsilon[0:]$	$\text{Sink}_0^\epsilon[1:]$	$\text{Sink}_0^\epsilon[2:]$	$\text{Sink}_0^\epsilon[4:]$	$\text{Sink}_0^\epsilon[8:]$	$\text{Sink}_0^\epsilon[16:]$
200704	Baseline	81.22	84.75	88.60	92.46	91.97	93.89
	Gated Attention	0.79	0.81	0.80	0.88	1.01	1.34
	Force-sink Masking	90.42	94.35	94.09	93.50	92.31	91.94
	Force-sink Masking$^+$	90.04	93.95	93.68	93.18	92.08	91.47
	TNLM Loss	69.20	72.21	75.49	83.04	91.89	90.34
	Force-sink Masking+TNLM Loss	90.72	94.66	94.42	93.86	92.57	92.36
	Force-sink Masking$^+$+TNLM Loss	89.83	93.74	93.45	92.80	91.87	92.09
202752	Baseline	81.53	85.07	88.94	92.83	92.36	94.30
	Gated Attention	0.85	0.87	0.88	0.97	1.11	1.51
	Force-sink Masking	90.45	94.39	94.13	93.54	92.37	92.10
	Force-sink Masking$^+$	90.16	94.08	93.81	93.32	92.25	91.73
	TNLM Loss	69.29	72.30	75.59	83.15	92.05	90.44
	Force-sink Masking+TNLM Loss	90.76	94.71	94.47	93.91	92.65	92.54
	Force-sink Masking$^+$+TNLM Loss	89.87	93.78	93.50	92.85	91.97	92.31
204800	Baseline	81.34	84.88	88.74	92.61	92.15	94.03
	Gated Attention	0.80	0.82	0.82	0.91	1.04	1.34
	Force-sink Masking	90.44	94.37	94.12	93.53	92.35	92.01
	Force-sink Masking$^+$	90.10	94.02	93.75	93.25	92.17	91.59
	TNLM Loss	69.21	72.22	75.50	83.05	91.97	90.25
	Force-sink Masking+TNLM Loss	90.75	94.69	94.45	93.90	92.63	92.44
	Force-sink Masking$^+$+TNLM Loss	89.89	93.80	93.52	92.87	91.98	92.27

Table 12: Sink scores at $\epsilon = 0.4$ for the last 3 checkpoints under 5× Chinchilla-optimal setting.