HYDRAHEAD: FROM HEAD-LEVEL FUNCTIONAL HETEROGENEITY TO SPECIALIZED ATTENTION HYBRIDIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

The quadratic complexity of attention poses a critical bottleneck for long-context processing, spurring interest in hybrid attention designs. Most open-source hybrid models adopt a layer-wise strategy. Yet, prior work has noted the inherent difficulty of integrating Linear Attention (LA) with Full Attention (FA), suggesting that the design space of attention hybridization remains underexplored. To probe this space, we conduct interpretability analysis and observe that layers exhibit block-wise functional similarity, while individual heads within the same layer display distinct functional specialization despite sharing input features. This head-level heterogeneity suggests that the head dimension provides a natural and principled granularity for fusing heterogeneous attention signals. Building on this insight, we introduce **HydraHead**, a novel architecture that hybridizes FA and LA along the head axis. HydraHead features two key innovations: (1) an interpretability-driven selection strategy that identifies retrieval-critical heads and preserves FA only for them, and (2) a scale-normalized fusion module that reconciles the distributional gap between FA and LA head outputs. By leveraging a three-stage transfer pipeline with parameter reuse and distillation, we achieve high-performance hybrid models with minimal training overhead. Under a unified training setup, HydraHead outperforms other hybrid designs in long-context tasks while maintaining strong general reasoning. With interpretability-driven head selection, it matches a 3:1 layer-wise hybrid's long-context performance at a 7:1 LA-to-FA ratio. Crucially, trained on only 15B tokens, HydraHead achieves over **69%** improvement over the baseline at 512K context length, approaching Qwen3.5, a leading model of comparable size with a native context length of 256K. This highlights the significant scaling potential of head-level hybridization.

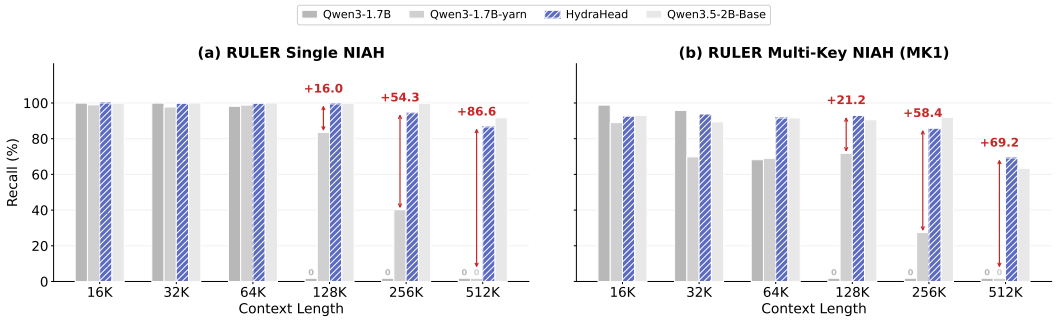


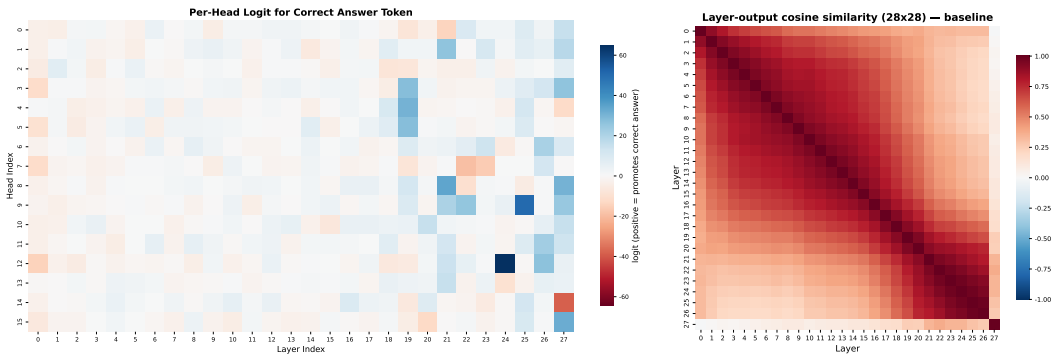
Figure 1: HydraHead, trained on only 15B tokens, significantly outperforms the Qwen3-1.7B baseline on NIAH tasks, approaching Qwen3.5-2B-Base, a leading model of comparable scale with native 256K support.

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.

1 INTRODUCTION

The landscape of Large Language Models (LLMs) has witnessed a pivotal transition from static question-answering systems to autonomous agents capable of complex reasoning and long-horizon planning (GLM-5-Team et al., 2026; Team et al., 2026b; Team, 2026; DeepSeek-AI et al., 2026). This evolution places unprecedented demands on context window extension, yet the quadratic complexity of standard Full Attention (FA) remains a rigid computational barrier. While Linear Attention (LA) paradigms—often rooted in State Space Models (SSMs) (Gu & Dao, 2023; Dao & Gu, 2024; Lahoti et al., 2026) or kernel-based approximations (Schlag et al., 2021; Yang et al., 2025b; Team et al., 2025b; Hatamizadeh et al., 2026)—offer linear-time complexity, they frequently suffer from "expressivity collapse," struggling to maintain the high-precision retrieval required for sophisticated reasoning tasks.

Driven by this tension, many recent LLMs adopt hybrid architectures—predominantly layer-wise designs that interleave different attention mechanisms across layers—integrating linear with softmax attention (Team, 2026; Team et al., 2025b; MiniMax et al., 2025), or blending full with sliding-window attention (Team et al., 2026a). These efforts reflect a shared direction: scaling long-context capabilities is no longer a matter of selecting a single attention mechanism, but an exercise in architectural fusion. Despite these successes, some prior work has noted that training hybrid models that combine linear attention with full attention remains challenging, with efforts ultimately focusing on full- or sparse-attention variants (GLM-5-Team et al., 2026; MiniMax et al., 2026; DeepSeek-AI et al., 2026), suggesting that the design space of attention hybridization remains underexplored.



(a) Per-head logit for correct answer token.

(b) Layer-output cosine similarity.

Figure 2: **Head-level variation motivates fine-grained allocation.** (a) Projected correct-answer logits vary among heads within the same layer in the retrieval analysis. (b) Layer-output cosine similarities show a smooth, block-wise structure across depth.

To navigate this underexplored design space, it is essential to look beyond surface-level performance and examine how computation is internally organized (Elhage et al., 2021; Wang et al., 2022; Gao et al., 2025; Templeton et al., 2026). Previous studies have identified specialized attention heads and substantial variation in their contributions to model behavior (Voita et al., 2019; Michel et al., 2019; Clark et al., 2019; Zhang et al., 2024; Chen et al., 2025). We examine this variation at both the head and layer levels. In our retrieval analysis, correct-answer logit readouts vary markedly among heads within the same layer (Figure 2a), whereas layer-output similarities exhibit a smooth, block-wise structure across depth (Figure 2b). Head-level analysis therefore exposes within-layer differences that a layer-wise assignment cannot accommodate. These observations motivate allocating attention mechanisms to individual heads; additional representational diagnostics are given in Appendix F.2.

Building on these insights, we propose **HydraHead**, a fine-grained head-level attention hybridization architecture that addresses two core challenges: (1) determining which heads should adopt FA versus LA, and (2) effectively fusing the heterogeneous attention patterns produced by different mechanisms. For the first challenge, we leverage interpretability techniques to identify heads critical for precise token-level retrieval directly from a pretrained model, retaining FA exclusively for these heads while assigning LA to the remainder for efficient long-context scaling. This head selection procedure is lightweight and one-shot, requiring only a few forward passes over a small calibration set. For

the second challenge, we mix FA and LA at the head dimension within each layer, assigning each attention head an independently configurable attention mechanism. Owing to the paradigm gap between FA and LA in how attention features are formed (Meng et al., 2026a;b), we introduce a head-wise scale-normalized fusion to mitigate interference when these heterogeneous attention signals are mixed. To train the resulting hybrid model, we introduce a three-stage transfer pipeline: parameter inheritance from the pretrained checkpoint and layer-wise alignment; global distillation; and long-context fine-tuning.

We evaluate HydraHead against several representative hybridization strategies, all built upon Qwen3-1.7B (Yang et al., 2025a) under a unified training configuration, using both long-context retrieval benchmarks at varying sequence lengths, and general reasoning benchmarks spanning multiple difficulty levels. These baselines span three dominant paradigms: layer-wise hybrids, token-wise hybrids, and head-wise hybrids. Our experiments yield the following findings.

- **HydraHead achieves state-of-the-art long-context performance while maintaining advanced general reasoning capabilities.** Existing hybridization strategies reveal a consistent trade-off across granularity levels: compared to layer-wise hybrids, both head-wise and token-wise approaches excel at general reasoning but struggle with length extrapolation. As a fine-grained head-wise approach, HydraHead retains a robust **>10%** gain over layer-wise baselines on hard reasoning tasks, while achieving **state-of-the-art** long-context performance.
- **Interpretability-based head selection enables aggressive FA compression with minimal performance degradation.** By effectively identifying retrieval-critical heads, our method matches the overall performance of a 3:1 layer-wise hybrid even at substantially higher LA-to-FA mixing ratios (e.g., 7:1).

Beyond controlled comparisons, we further challenge HydraHead by directly benchmarking against state-of-the-art models. As shown in Figure 1, through meticulous optimization of training hyperparameters and data composition, our model achieves substantial long-context capabilities with only 15B training tokens. Specifically, it delivers over **69%** improvement on NIAH benchmarks at 512k context compared to the pretrained baseline, approaching the performance of Qwen3.5 (Team, 2026), a leading model of comparable scale. Crucially, these gains are achieved without compromising general reasoning performance, further validating the significant potential of our head-wise hybrid method in efficient long-context modeling.

2 METHOD

We take GQA (Ainslie et al., 2023) and Gated DeltaNet (GDN) (Yang et al., 2025b) as the representatives of FA and LA, respectively. In a layer with H query heads, each head's output is either $\mathbf{O}_h^F$ from the GQA or $\mathbf{O}_h^L$ from the GDN. Full preliminaries are deferred to Appendix A.2 A.3.

Next, we describe how to select critical heads (Section 2.1), our hybrid structure (Section 2.2), and how to transfer-train from a pretrained model (Section 2.3).

2.1 HEAD IMPORTANCE ESTIMATION VIA CAUSAL INTERVENTION

Deciding which heads should retain full attention, and which can be safely approximated by linear recurrence, is the decision on which our efficiency–quality trade-off hinges. The head-level variation in Figure 2 motivates this selective allocation. We turn it into a ranking by measuring how interventions on individual heads change the target behavior using activation and path patching (Appendix A.1).

Behavioral readout. For the target behavior, we construct calibration pairs (x, x') in which the counterfactual input changes behavior-relevant information while preserving the surrounding structure. A scalar readout m measures evidence for the clean target behavior as specified in Section 3.1.1; calibration and candidate-screening details are given in Appendix F.1.

Receiver effects. For a candidate query head h in layer l, we replace its clean output $\mathbf{O}_{l,h}(x)$ with its counterfactual output $\mathbf{O}_{l,h}(x')$. Downstream attention outputs remain fixed to their clean values, so the intervention reaches the readout through the residual stream and MLP computation.

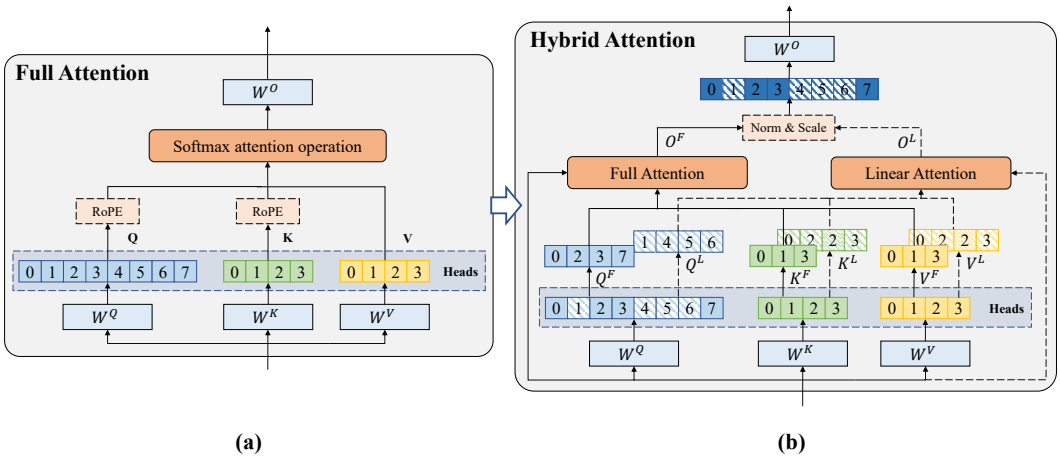


Figure 3: Comparison of standard FA and our proposed head-wise hybrid attention. A subset of heads retains the original FA branch, while the remaining heads employ GDN as a representative LA variant to reduce computational overhead.

We normalize its effect by the clean–counterfactual readout difference:

$$\mathrm{IE}_{l,h}(x, x') = \frac{m(x) - m(x;\, \mathbf{O}_{l,h} \leftarrow \mathbf{O}_{l,h}(x'))}{m(x) - m(x')}. \tag{1}$$

Averaging over calibration pairs gives the receiver score $\mathrm{IE}^{\mathrm{recv}}_{l,h}$. High-scoring heads form the receiver set for tracing upstream contributions.

Sender effects. Heads can also support the target behavior by supplying information to a receiver. For each candidate upstream head, path patching restricts its intervention to the selected receiver pathways within an otherwise-clean execution. The normalized readout decrease, averaged over calibration pairs, gives the sender score $\mathrm{IE}^{\mathrm{send}}_{l,h}$. We repeat the tracing procedure with newly identified senders acting as receivers in the next round to capture upstream contributions.

Head allocation. We combine the two roles using max pooling, weighted by consistency:

$$S_{l,h} = \max\left(\mathrm{IE}^{\mathrm{recv}}_{l,h}, \mathrm{IE}^{\mathrm{send}}_{l,h}\right) \kappa_{l,h}, \tag{2}$$

where $\kappa_{l,h}$ is the fraction of probes on which the head's larger receiver or sender effect reaches a threshold τ. We rank candidate heads globally by $S_{l,h}$ and retain FA for the top-K heads, with K set by the FA budget. The remaining heads use LA. This assignment can retain different numbers of FA heads in different layers and determines the branch partition implemented in Section 2.2.

2.2 Head-wise Hybridization

FA enables precise token-level retrieval but scales quadratically with sequence length, whereas LA achieves $O(T)$ long-context processing at the cost of compressing history into a fixed-size state, degrading recall fidelity over extended contexts. To harness both strengths, we propose a *Head-wise Hybrid Attention* mechanism (Figure 3): each query head is assigned to either the FA or the LA (GDN) branch according to its functional importance. This fine-grained hybridization retains critical reasoning in specific FA heads while exploiting the $O(T)$ efficiency and stable state evolution of GDN for effective context extension.

Head Partitioning. Let $\mathcal{H} = \{1, \ldots, H\}$ denote the set of all query head indices. We partition $\mathcal{H}$ into two disjoint subsets: $\mathcal{H}_F$ for heads assigned to the FA, and $\mathcal{H}_L$ for heads assigned to the GDN branch, such that $\mathcal{H}_F \cup \mathcal{H}_L = \mathcal{H}$ and $\mathcal{H}_F \cap \mathcal{H}_L = \emptyset$. The assignment is determined via an interpretability-driven selection process (detailed in Section 2.1), which identifies heads crucial for precise context retrieval and complex reasoning.

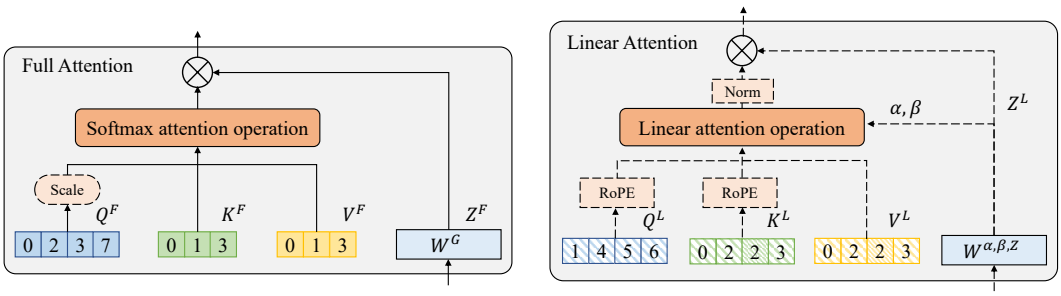


Figure 4: Details of FA and LA branches. We use GDN as a representative linear attention variant to illustrate the design. Specifically, we replace the original RoPE in the FA branch with a scaling mechanism, while applying RoPE to the LA (GDN) branch to enhance its positional awareness.

Parallel Branch Computation. Each head is routed to its assigned branch: heads in $\mathcal{H}_F$ produce outputs $\mathbf{O}_h^F$ via standard softmax attention, while heads in $\mathcal{H}_L$ produce $\mathbf{O}_h^L$ via the Gated DeltaNet recurrence.

Head-wise Scale-normalized Fusion. As shown in NaLaFormer (Meng et al., 2026b), softmax encodes query magnitude into FA scores, producing sharp, low-entropy attention, whereas LA's normalization cancels query magnitude, producing smooth, high-entropy attention. This asymmetry propagates to outputs: FA heads yield query-norm-modulated features dominated by few high-weight tokens, while LA heads yield uniform, query-norm-agnostic features, so naïvely concatenating the two streams destabilizes optimization.[1] We therefore propose independent normalization and head-wise modulation, reconciling the distributional gap while preserving the structural integrity of the original heads.

Let $\mathbf{O}_h^F \in \mathbb{R}^{T\times d_h}$ and $\mathbf{O}_h^L \in \mathbb{R}^{T\times d_h}$ denote the raw outputs of the FA and GDN branches for head h, respectively. We first apply RMSNorm independently to each head's output to unify their feature scales:

$$\hat{\mathbf{O}}_h = \mathrm{Norm}(\mathbf{O}_h), \quad \text{where } \mathbf{O}_h = \begin{cases} \mathbf{O}_h^F & \text{if } h \in \mathcal{H}_F, \\ \mathbf{O}_h^L & \text{if } h \in \mathcal{H}_L. \end{cases} \tag{3}$$

Crucially, we concatenate these normalized head outputs $\{\hat{\mathbf{O}}_1, \ldots, \hat{\mathbf{O}}_H\}$ along the head dimension *according to their original head indices* to form a tensor $\hat{\mathbf{O}} \in \mathbb{R}^{T\times H\times d_h}$. This index-preserving concatenation ensures that the positional and functional identity of each head is maintained within the global representation.

To fully utilize the characteristics of these two attention mechanisms, we introduce a learnable head-wise scaling vector $\boldsymbol{\gamma} \in \mathbb{R}^H$. This vector assigns a scalar weight to each head based on its original index, allowing the model to adaptively recalibrate the contribution of each head regardless of its underlying attention type. The modulated tensor $\tilde{\mathbf{O}} \in \mathbb{R}^{T\times H\times d_h}$ is computed via broadcasting multiplication along the head dimension:

$$\tilde{\mathbf{O}}_{:,h,:} = \gamma_h \cdot \hat{\mathbf{O}}_{:,h,:}, \quad \forall h \in [1, H]. \tag{4}$$

Finally, $\tilde{\mathbf{O}}$ is reshaped into $\mathbb{R}^{T\times d}$ (where $d = H \cdot d_h$) and projected by the shared output matrix $\mathbf{W}^O$ to produce the final attention output. This design ensures that the distributional gaps between FA and GDN features are bridged before fusion, while preserving the distinct functional roles of individual heads through learnable scalars aligned with their original positions.

Branch-specific refinements. We apply attention-specific refinements to each branch (Figure 4). For the FA branch, we remove RoPE and apply a log-scale coefficient to queries for stable long-context attention, and add an auxiliary gate to mitigate attention sinks. For the GDN branch, we integrate RoPE into query/key projections to improve positional awareness, and expand key-value heads to match query heads. Details and motivations are deferred to Appendix B.1.

[1] We observe that when FA and LA are unevenly distributed, direct concatenation significantly degrades capabilities (Table 13).

2.3 Efficient Hybrid Transfer Learning

Transforming a pre-trained FA model into a hybrid architecture reduces training overhead while preserving strong performance. The key challenge is that the newly introduced GDN branches start with different computational dynamics than the FA heads they replace, so direct fine-tuning is unstable. Inspired by prior work on hybrid distillation (Li et al., 2025; Chen et al., 2026), we address this through a three-stage pipeline with a principled progression: first align local behavior (Stage 1), then align global semantics (Stage 2), and finally adapt to the target domain (Stage 3).

Stage 1 freezes the backbone and aligns each hybrid attention layer's hidden state with the original FA layer via MSE. Stage 2 unfreezes the full model and distills the teacher's output distribution via KL divergence. Stage 3 performs standard supervised fine-tuning with next-token prediction on longer contexts. Implementation details are provided in Appendix B.2.

3 Experiments

We evaluate HydraHead along four questions: (1) **Does it outperform existing hybrid designs?** (Section 3.2); (2) **Does interpretability-guided selection help?** (Section 3.3); (3) **Is scale-normalized fusion effective?** (Section 3.4); and (4) **How far can it scale?** (Section 3.5). Additional ablations and analyses are provided in Appendix G.

3.1 Experimental Setup

3.1.1 Model, Training, and Head Selection Configuration

We initialize from Qwen3-1.7B (Yang et al., 2025a) and train on FineWeb-Edu (Penedo et al., 2024), retaining FA in 25% of heads and converting the rest to GDN. Critical FA heads are selected via retrieval calibration on five Needle-in-a-Haystack probes from RULER (Hsieh et al., 2024); training follows the three-stage pipeline of Section 2.3. Head-selection details and the full training schedule are in Appendix F.1 and Table 6, respectively.

3.1.2 Evaluation benchmarks

To comprehensively assess the effectiveness of our method, we evaluate the model in two primary dimensions: **Long-Context Retrieval Capability** (Needle In A Haystack) and **General Reasoning Proficiency**. For retrieval, we use RULER (Hsieh et al., 2024) at 16K–256K tokens and report native-context (16K/32K) and extended-context (64K/128K/256K) averages. For general reasoning, we report standard academic (Easy) and challenging benchmarks (Hard) evaluated with lm-evaluation-harness (Gao et al., 2024). The full benchmark list and protocol are given in Appendix C.2.

3.2 Hybrid Architecture Comparison

We first systematically compare HydraHead against representative hybridization baselines spanning layer-wise, token-wise, and head-wise paradigms under the transfer learning protocol described in Section 2.3, to elucidate its distinct advantages.

Experimental setup. We use GDN as the linear attention mechanism and compare three hybridization paradigms. **Layer-wise** baselines follow HypeNet (Chen et al., 2026): 25% FA layers plus a GDN remainder, and a uniform variant. **Token-wise** baselines combine 4K-window SWA with GDN in every layer, or use a Liger-style interleaved design (75% SWA+GDN with window 128, 25% FA+GDN). **Head-wise mixing** uses Hymba's parallel FA–GDN fusion over all heads. For **head-wise selection**, we use our HydraHead, retaining 25% of heads as FA. Details are in Appendix D.

Results and analysis. Table 1 compares HydraHead against representative layer-wise, token-wise, and head-wise hybrids under identical training conditions and parameter budgets. Token- and head-wise hybrids outperform layer-wise hybrids on difficult general reasoning by over 10%, which may reflect that fine-grained within-layer interaction better preserves the complex dependencies that layer-wise substitution disrupts. Layer-wise hybrids, however, better extend context length.

Table 1: Architecture comparison of various hybrid designs. Cache denotes the relative key-value cache under a 16K context length, where the standard FA cache is normalized to 1.0. Light. denotes the lightning attention (Qin et al., 2024). * and ‡ indicate the hybrids are similar to HypeNet (Chen et al., 2026) and Liger (Lan et al., 2025), respectively. † indicates that an FA layer is inserted after every 3 LA layers. More details are provided in Appendix D.

Model	LA	Cache	RULER Single		RULER Multi-Key		General Reasoning	
			(Native)	(Extended)	(Native)	(Extended)	(Hard)	(Easy)
Layer-wise Hybrid Transformers								
FA & LA*	Light.	0.25	89.07	85.00	24.85	24.37	19.80	59.72
FA & LA†	Light.	0.25	86.03	67.76	31.70	17.80	15.94	60.29
FA & LA	GDN	0.25	47.63	7.02	22.50	5.53	16.44	58.18
LA-FA & LA	GDN	0.25	58.50	8.13	19.00	6.83	18.42	58.70
Token-wise & Head-wise Hybrid Transformers								
Token-wise	GDN	0.25	20.77	3.73	16.05	2.43	**47.31**	63.40
Token-wise‡	GDN	0.25	96.03	58.78	26.20	11.03	36.66	**63.37**
Head-wise Mixing	GDN	1.00	93.20	60.42	**37.35**	14.53	38.07	62.77
HydraHead	GDN	0.35	**98.47**	**87.49**	37.10	**27.37**	31.03	62.12
HydraHead (7:1)	GDN	0.20	88.70	81.04	26.70	22.93	29.46	62.65

HydraHead breaks this trade-off: interpretability-guided selection preserves the critical FA heads for global context awareness, yielding state-of-the-art long-context extrapolation while maintaining robust general-domain capabilities. Furhtermore, with higher 1:7 hybrid ratios, our model matches the overall performance of the 1:3 layer-wise hybrid (* in Table 1). More details are in Appendix G.3. We note that the pure SWA+GDN variant, without FA-interleaved layers, performs poorly on long contexts, possibly because the current transfer learning setup struggles to generalize beyond $4\times$ the training window.

3.3 Effectiveness of Interpretability Screening

To validate the hypothesis that functional heterogeneity among attention heads drives the efficacy of hybrid architectures, we investigate the impact of head selection strategies.

Experimental Setup. We compare five distinct head allocation strategies, all maintaining a global fixed ratio of 3:1 between GDN and FA heads to ensure comparable computational budgets. The baselines include: **(a) Fixed Allocation**, which uniformly assigns a fixed fraction of heads to FA in every layer; **(b) Layer-wise Random Selection**, which randomly selects FA heads within each layer; **(c) Global Random Selection**, which randomly selects FA heads across the entire network. As our proposed methods, we evaluate two interpretability-driven variants based on causal importance scores: **(d) Layer-wise Interpretability Screening**, which selects the top-ranked heads within each layer according to their local importance; **(e) Global Interpretability Screening**, which selects the most critical heads across all layers regardless of their positional distribution. Figure 10 visualizes the resulting head distributions for these five strategies, highlighting the non-uniform, layer-skewed patterns emerging from the global interpretability-based approach.

Results and Analysis. The comparative results across the five selection strategies are summarized in Table 2. We have the following findings:

Baseline stability and global random degradation. Fixed allocation and layer-wise random selection are nearly identical across all evaluation dimensions, suggesting uniform FA-head distribution is stable, and random within-layer variation does not significantly alter model capacity. In contrast, global random selection degrades both single-key and multi-key retrieval, indicating that disrupting layer-specific balance without functional guidance harms cross-depth information coordination.

Interpretability screening. Layer-wise interpretability screening improves performance across nearly all benchmarks by prioritizing causally important heads within each layer, validating that preserving critical heads in the FA branch outperforms naive uniform allocation. Global screening

Table 2: Comparison of different head selection strategies. All experiments are conducted based on **Basic Model**.

Model	RULER Single		RULER Multi-Key		General Reasoning	
	(Native)	(Extended)	(Native)	(Extended)	(Hard)	(Easy)
(a) Fixed	85.63	62.62	27.35	13.67	28.65	62.59
(b) Layer-Rand	87.07	65.33	27.70	17.37	27.91	**62.95**
(c) Global-Rand	59.40	32.96	16.55	5.33	26.19	62.40
(d) Layer-Interp	91.13	65.22	35.10	15.97	**31.93**	62.15
(e) Global-Interp	**98.70**	**81.73**	**38.35**	**25.93**	31.70	62.68

Table 3: Comparison between direct concatenation and the proposed scale-normalized fusion.

Fusion Method	RULER Single		RULER Multi-Key		General Reasoning	
	(Native)	(Extended)	(Native)	(Extended)	(Hard)	(Easy)
Direct Concatenation	85.07	71.36	**41.90**	23.93	30.62	**62.58**
Scale-normalized Fusion	**98.47**	**87.49**	37.10	**27.37**	**31.03**	62.12

further unlocks potential by selecting the most important heads across the entire network, enabling a non-uniform allocation that concentrates FA resources in the layers most critical for reasoning and recall (Figure 10). This sparse but effective pattern indicates that functional heterogeneity drives efficient hybrid attention design.

Why interpretability outperforms naive strategies. Attention heads vary dramatically in importance (Voita et al., 2019; Michel et al., 2019), and our localization shows only $\approx$6.5% are causally critical for retrieval (Appendix F.3). Fixed allocation wastes FA budget on unimportant heads, while global random selection can leave entire layers without FA (Table 2(c)). Interpretability screening directs every FA slot to retrieval-critical heads, dominating especially aggressive ratios (Table 14).

3.4 Feature Fusion

We investigate whether the proposed fusion module can robustly integrate the heterogeneous features produced by FA and GDN heads.

Experimental Setup. Under interpretability-guided global head selection, we compare our scale-normalized fusion with a naive baseline that directly concatenates the raw FA and GDN features.

Results and Analysis. As shown in Table 3, scale-normalized fusion improves native and extended Single-key retrieval by 13.40 and 16.13 points, respectively, and extended Multi-Key retrieval by 3.44 points, while maintaining comparable general reasoning performance. Although direct concatenation performs better on native Multi-Key retrieval, it degrades substantially under context extension. These results demonstrate that our fusion robustly integrates heterogeneous features, particularly for long-context generalization. More analyses are provided in Appendix G.2.

3.5 Scaling Up with More Training Tokens

We scale the high-quality training data to over 15B tokens and benchmark HydraHead against open-source 2B-parameter models, including standard Transformers and hybrid architectures.

Experimental setup. All baselines are official releases evaluated under a unified framework with identical preprocessing and hyperparameters. We report NIAH single-key and multi-key averages from 4K to 256K tokens for long-context retrieval, and the hard subset for general reasoning. Comparisons include standard Transformers (Qwen3-1.7B, MiniCPM-1B-Base (Hu et al., 2024)) and hybrid architectures (Gemma-3n-E2B (Team et al., 2025a), HypeNet, Hymba-1.5B (Dong et al., 2024), Jet-Nemotron-2B (Gu et al., 2025)).

Table 4: Comparison with SOTA models on long-context retrieval benchmarks across context lengths ranging from 4K to 256K. **Bold** indicates the best result, and underlining indicates the second best.

Model	Type	4K	8K	16K	32K	64K	128K	256K
RULER Single & Multi-Key								
Qwen3-1.7B	FA	**99.75**	**99.64**	**98.87**	**97.02**	70.16	0.00	0.00
Qwen3-1.7B-yarn	FA	98.70	95.87	76.62	71.92	73.30	62.49	27.20
MiniCPM5-1B-Base	FA	99.02	98.04	94.25	92.40	83.57	74.07	38.07
Gemma-3n-E2B	Hybrid	99.32	99.19	94.69	87.99	16.97	0.00	0.00
HypeNet-2B	Hybrid	62.05	61.72	55.67	49.12	55.67	47.4	42.92
Hymba-1.5B	Hybrid	69.85	60.8	19.05	0.04	0.00	0.00	0.00
Jet-Nemotron-2B	Hybrid	97.85	94.79	84.24	67.40	18.04	7.25	0.54
HydraHead	Hybrid	99.25	98.17	97.15	96.15	**90.90**	**84.14**	**73.62**

Table 5: Comparison with SOTA models on the general benchmark (**Hard**).

Model	Type	MMLU	BBH	MBPP	GSM8k	Average
Qwen3-1.7B	FA	55.46	48.16	43.00	69.45	54.02
Qwen3-1.7B-yarn	FA	53.45	45.65	42.40	59.97	50.37
MiniCPM5-1B-Base	FA	39.30	37.81	33.20	34.12	36.11
Gemma-3n-E2B	Hybrid	53.82	42.47	35.00	25.85	39.29
HypeNet-2B	Hybrid	46.52	9.15	4.80	4.93	16.35
Hymba-1.5B	Hybrid	49.67	8.89	16.60	0.99[2]	19.04
Jet-Nemotron-2B	Hybrid	53.59	**58.16**	**54.20**	**75.28**	**60.31**
HydraHead	Hybrid	**55.80**	46.64	47.20	52.84	50.62

Results and Analysis. We present quantitative results on long-context retrieval (Table 4) and general reasoning (Table 5), and summarize our findings in two aspects below.

Superior long-context extrapolation over standard Transformers. HydraHead extrapolates strongly beyond 64K tokens, surpassing the YaRN-enhanced Qwen3-1.7B by +46.42% on RULER at 256K. Within the native 32K range, it remains competitive with the base model and exceeds YaRN on some tasks. On general reasoning, HydraHead trails Qwen3-1.7B by only 3.40 percentage points—a small gap given only 15B tokens of transfer training—and matches the YaRN variant, indicating that the hybrid architecture does not compromise general capabilities.

Robustness advantage over existing hybrid architectures. While competing hybrids degrade to near-zero at 256K, HydraHead sustains 73.62%. On general reasoning, it outperforms all other hybrids by at least 10 percentage points on average, except Jet-Nemotron-2B (+9.69). However, Jet-Nemotron-2B sacrifices long-context ability, trailing HydraHead by over 70 percentage points at $\geq$64K. This validates that our fine-grained head-wise allocation preserves strong general reasoning while ensuring superior long-context scalability, offering a more balanced solution than current hybrid models.

4 Conclusion

In this paper, we introduce HydraHead, a head-wise hybrid attention architecture that assigns interpretability-identified critical heads to FA and the remainder to LA. Scale-normalized fusion reconciles their heterogeneous features, while a three-stage pipeline combining parameter reuse and distillation enables efficient transfer from pretrained Transformers.

Under controlled settings, HydraHead improves challenging reasoning performance by over 10% compared with layer-wise hybrids and consistently performs better on long-context tasks. Trained on only 15B tokens, it also surpasses the pretrained baseline, demonstrating strong scaling potential.

[2]We evaluate the Hymba-1.5B base model, for which no official result is reported.

AI USE STATEMENT

In this work, we used generative AI tools to assist with implementing the research code used to realize the methods described in this paper. We did not use generative AI tools for other tasks requiring disclosure, and the remaining required-disclosure tasks are not applicable to this work. Additionally, we used generative AI tools to improve the clarity and readability of the manuscript. We reviewed all AI-assisted work: all code suggestions were manually inspected, tested, and validated, and all text revisions were checked for accuracy and fidelity to the intended content. We take full responsibility for the final content of this work, including any text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

All experiments are conducted under a unified training and evaluation pipeline. The base model (Qwen3-1.7B) and all baseline models are official releases from their respective authors. Evaluation is performed using the lm-evaluation-harness framework with default configurations. The head importance estimation procedure is described in detail in Section 2.1, with calibration set configuration specified in Appendix F.3 and Appendix F.1. Training configurations for all three stages are provided in Tables 6 and 16. Detailed architectural specifications for all hybrid baselines are provided in Appendix D. Code and trained checkpoints will be released upon acceptance.

ETHICS STATEMENT

This work focuses on architectural improvements for efficient attention mechanisms in large language models. Our research does not involve human subjects, sensitive data collection, or applications that raise immediate ethical concerns. The training data (FineWeb-Edu) is a publicly available, filtered educational corpus released under the Open Data Commons Attribution License (ODC-By). We believe that improving the efficiency of long-context processing can contribute positively to democratizing access to capable language models by reducing computational requirements.

REFERENCES

Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebrón, and Sumit Sanghai. Gqa: Training generalized multi-query transformer models from multi-head checkpoints, 2023. URL https://arxiv.org/abs/2305.13245.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. Program synthesis with large language models, 2021. URL https://arxiv.org/abs/2108.07732.

Aviv Bick, Kevin Y. Li, Eric P. Xing, J. Zico Kolter, and Albert Gu. Transformers to ssms: Distilling quadratic knowledge to subquadratic models, 2025. URL https://arxiv.org/abs/2408.10189.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. Piqa: Reasoning about physical commonsense in natural language, 2019. URL https://arxiv.org/abs/1911.11641.

Wei Chen, Zhen Huang, Liang Xie, Binbin Lin, Houqiang Li, Le Lu, Xinmei Tian, Deng Cai, Yonggang Zhang, Wenxiao Wang, Xu Shen, and Jieping Ye. From yes-men to truth-tellers: Addressing sycophancy in large language models with pinpoint tuning, 2025. URL https://arxiv.org/abs/2409.01658.

Yingfa Chen, Zhen Leng Thai, Zihan Zhou, Zhu Zhang, Xingyu Shen, Shuo Wang, Chaojun Xiao, Xu Han, and Zhiyuan Liu. Hybrid linear attention done right: Efficient distillation and effective architectures for extremely long contexts, 2026. URL https://arxiv.org/abs/2601.22156.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. What does bert look at? an analysis of bert's attention, 2019. URL https://arxiv.org/abs/1906.04341.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge, 2018. URL https://arxiv.org/abs/1803.05457.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

Tri Dao and Albert Gu. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. In *International Conference on Machine Learning (ICML)*, 2024.

DeepSeek-AI, Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chengyu Hou, Chenhao Xu, Chenze Shao, Chong Ruan, Conner Sun, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Donghao Li, Dongjie Ji, Erhang Li, Fang Wei, Fangyun Lin, Fangzhou Yuan, Feiyu Xia, Fucong Dai, Guangbo Hao, Guanting Chen, Guoai Cao, Guolai Meng, Guowei Li, Han Yu, Han Zhang, Hanwei Xu, Hao Li, Haofen Liang, Haoling Zhang, Haoming Luo, Haoran Wei, Haotian Yuan, Haowei Zhang, Haowen Luo, Haoyu Chen, Haozhe Ji, Hengqing Zhang, Honghui Ding, Hongxuan Tang, Huanqi Cao, Huazuo Gao, Hui Qu, Hui Zeng, J Yang, JQ Zhu, Jia Luo, Jia Song, Jia Yu, Jialiang Huang, Jialu Cai, Jian Liang, Jiangting Zhou, Jiasheng Ye, Jiashi Li, Jiaxin Xu, Jiewen Hu, Jieyu Yang, Jin Chen, Jin Yan, Jingchang Chen, Jingli Zhou, Jingting Xiang, Jingyang Yuan, Jingyuan Cheng, Jingzi Zhou, Jinhua Zhu, Jiping Yu, Joseph Sun, Jun Ran, Junguang Jiang, Junjie Qiu, Junlong Li, Junmin Zheng, Junxiao Song, Kai Dong, Kaige Gao, Kang Guan, Kexing Zhou, Kezhao Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Wang, Leyi Xia, Li Zhang, Liang Zhao, Lihua Guo, Lingxiao Luo, Linwang Ma, Linyan Zhu, Litong Wang, Liyu Cai, Liyue Zhang, Longhao Chen, MS Di, MY Xu, Max Mei, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Mingxu Zhou, Minmin Han, Ning Wang, Panpan Huang, Panpan Wang, Peixin Cong, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qingyang Li, Qinyu Chen, Qiushi Du, Qiwei Jiang, Rui Tian, Ruifan Xu, Ruijie Lu, Ruiling Xu, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runqian Chen, Runqiu Yin, Runxin Xu, Ruomeng Shen, Ruoyu Zhang, Ruyi Chen, SH Liu, Shanghao Lu, Shangmian Sun, Shangyan Zhou, Shanhuang Chen, Shaofei Cai, Shaoheng Nie, Shaoqing Wu, Shaoyuan Chen, Shengding Hu, Shengyu Liu, Shiqiang Hu, Shirong Ma, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, Shuying Yu, Songyang Zhou, Tao Ni, Tao Yun, Tian Jin, Tian Pei, Tian Ye, Tianle Lin, Tianran Ji, Tianyi Cui, Tianyuan Yue, Tingting Yu, Tun Wang, W Zhang, WL Xiao, Wangding Zeng, Wei An, Weilin Zhao, Wen Liu, Wenfeng Liang, Wenjie Pang, Wenjing Luo, Wenjing Yao, Wenjun Gao, Wenkai Yang, Wenlve Huang, Wenqing Hou, Wentao Zhang, Wenting Ma, Xi Gao, Xiang He, Xiangwen Wang, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaokang Zhang, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingchen Liu, Xingkai Yu, Xingyou Li, Xinyu Yang, Xinyu Zhang, Xu Chen, Xuanyu Wang, Xuecheng Su, Xueyin Chen, Xuheng Lin, Xuwei Fu, YC Yan, YQ Wang, YW Ma, Yanfeng Luo, Yang Zhang, Yanhong Xu, Yanru Ma, Yanwen Huang, Yao Li, Yao Li, Yao Xu, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Qian, Yi Shao, Yi Yu, Yichao Zhang, Yifan Ding, Yifan Shi, Yijia Wu, Yiliang Xiong, Yiling Ma, Ying He, Ying Tang, Ying Zhou, Yingjia Luo, Yinmin Zhong, Yishi Piao, Yisong Wang, Yixiang Zhang, Yixiao Chen, Yixuan Tan, Yixuan Wei, Yiyang Ma, Yiyuan Liu, Yonglun Yang, Yongqiang Guo, Yongtong Wu, Yu Wu, YuKun Li, Yuan Cheng, Yuan Ou, Yuanfan Xu, Yuanhao Li, Yuduan Wang, Yuehan Yang, Yuer Xu, Yuhan Wu, Yuhao Meng, Yuheng Zou, Yukun Zha, Yunfan Xiong, Yupeng Chen, Yuping Lin, Yuqian Cao, Yuqian Wang, Yushun Zhang, Yuting Yan, Yutong Lin, Yuxian Gu, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuxuan Zhou, Yuyang Zhou, Yuzhen Huang, ZF Wu, Zehao Wang, Zehua Zhao, Zehui Ren, Zekai Zhang, Zhangli Sha, Zhe Fu, Zhe Ju, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zheren Gao, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhixian Huang, Zhixuan Chen, Zhiyu Wu, Zhizhou Ren, Zhongyu Wu, Zhuoshu Li, Zhuping Zhang, Zian Xu, Zihao Wang, Zihua Qu, Zihui Gu, Zijia Zhu, Zilin Li, Zipeng Zhang, Ziwei Xie, Ziyi Gao, Ziyi Wan, Zizheng Pan, and Zongqing Yao. Deepseek-v4: Towards highly efficient million-token context intelligence, 2026. URL https://arxiv.org/abs/2606.19348.

Xin Dong, Yonggan Fu, Shizhe Diao, Wonmin Byeon, Zijia Chen, Ameya Sunil Mahabaleshwarkar, Shih-Yang Liu, Matthijs Van Keirsbilck, Min-Hung Chen, Yoshi Suhara, Yingyan Lin, Jan Kautz,

and Pavlo Molchanov. Hymba: A hybrid-head architecture for small language models, 2024. URL https://arxiv.org/abs/2411.13676.

Jusen Du, Jiaxi Hu, Tao Zhang, Weigao Sun, and Yu Cheng. Native hybrid attention for efficient sequence modeling, 2026. URL https://arxiv.org/abs/2510.07019.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tristan Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Scott Laskin, Scott Lee, Duane Lindner, Matt Maskey, Parker Nicholson, Karina Nguyen, Ethan Perez, Sam Ringer, Craig Sanderson, Toby Shevlane, Miles Tran, Lily Weng, Jeff Wu, Michael Ziegler, and Chris Olah. A mathematical framework for transformer circuits. https://transformer-circuits.pub/2021/framework/index.html, 2021. Anthropic Research.

Yonggan Fu, Xin Dong, Shizhe Diao, Matthijs Van Keirsbilck, Hanrong Ye, Wonmin Byeon, Yashaswi Karnati, Lucas Liebenwein, Han Zhang, Nikolaus Binder, Maksim Khadkevich, Alexander Keller, Jan Kautz, Yingyan Lin, and Pavlo Molchanov. Nemotron-flash: Towards latency-optimal hybrid small language models. *ArXiv*, abs/2511.18890, 2025. URL https://api.semanticscholar.org/CorpusID:283243510.

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.

Leo Gao, Achyuta Rajaram, Jacob Coxon, Soham V. Govande, Bowen Baker, and Dan Mossing. Weight-sparse transformers have interpretable circuits. *CoRR*, abs/2511.13653, 2025.

Yoav Gelberg, Koshi Eguchi, Takuya Akiba, and Edoardo Cetin. Extending the context of pretrained llms by dropping their positional embeddings, 2025. URL https://arxiv.org/abs/2512.12167.

GLM-5-Team, :, Aohan Zeng, Xin Lv, Zhenyu Hou, Zhengxiao Du, Qinkai Zheng, Bin Chen, Da Yin, Chendi Ge, Chenghua Huang, Chengxing Xie, Chenzheng Zhu, Congfeng Yin, Cunxiang Wang, Gengzheng Pan, Hao Zeng, Haoke Zhang, Haoran Wang, Huilong Chen, Jiajie Zhang, Jian Jiao, Jiaqi Guo, Jingsen Wang, Jingzhao Du, Jinzhu Wu, Kedong Wang, Lei Li, Lin Fan, Lucen Zhong, Mingdao Liu, Mingming Zhao, Pengfan Du, Qian Dong, Rui Lu, Shuang-Li, Shulin Cao, Song Liu, Ting Jiang, Xiaodong Chen, Xiaohan Zhang, Xuancheng Huang, Xuezhen Dong, Yabo Xu, Yao Wei, Yifan An, Yilin Niu, Yitong Zhu, Yuanhao Wen, Yukuo Cen, Yushi Bai, Zhongpei Qiao, Zihan Wang, Zikang Wang, Zilin Zhu, Ziqiang Liu, Zixuan Li, Bojie Wang, Bosi Wen, Can Huang, Changpeng Cai, Chao Yu, Chen Li, Chengwei Hu, Chenhui Zhang, Dan Zhang, Daoyan Lin, Dayong Yang, Di Wang, Ding Ai, Erle Zhu, Fangzhou Yi, Feiyu Chen, Guohong Wen, Hailong Sun, Haisha Zhao, Haiyi Hu, Hanchen Zhang, Hanrui Liu, Hanyu Zhang, Hao Peng, Hao Tai, Haobo Zhang, He Liu, Hongwei Wang, Hongxi Yan, Hongyu Ge, Huan Liu, Huanpeng Chu, Jia'ni Zhao, Jiachen Wang, Jiajing Zhao, Jiamin Ren, Jiapeng Wang, Jiaxin Zhang, Jiayi Gui, Jiayue Zhao, Jijie Li, Jing An, Jing Li, Jingwei Yuan, Jinhua Du, Jinxin Liu, Junkai Zhi, Junwen Duan, Kaiyue Zhou, Kangjian Wei, Ke Wang, Keyun Luo, Laiqiang Zhang, Leigang Sha, Liang Xu, Lindong Wu, Lintao Ding, Lu Chen, Minghao Li, Nianyi Lin, Pan Ta, Qiang Zou, Rongjun Song, Ruiqi Yang, Shangqing Tu, Shangtong Yang, Shaoxiang Wu, Shengyan Zhang, Shijie Li, Shuang Li, Shuyi Fan, Wei Qin, Wei Tian, Weining Zhang, Wenbo Yu, Wenjie Liang, Xiang Kuang, Xiangmeng Cheng, Xiangyang Li, Xiaoquan Yan, Xiaowei Hu, Xiaoying Ling, Xing Fan, Xingye Xia, Xinyuan Zhang, Xinze Zhang, Xirui Pan, Xu Zou, Xunkai Zhang, Yadi Liu, Yandong Wu, Yanfu Li, Yidong Wang, Yifan Zhu, Yijun Tan, Yilin Zhou, Yiming Pan, Ying Zhang, Yinpei Su, Yipeng Geng, Yong Yan, Yonglin Tan, Yuean Bi, Yuhan Shen, Yuhao Yang, Yujiang Li, Yunan Liu, Yunqing Wang, Yuntao Li, Yurong Wu, Yutao Zhang, Yuxi Duan, Yuxuan Zhang, Zezhen Liu, Zhengtao Jiang, Zhenhe Yan, Zheyu Zhang, Zhixiang Wei, Zhuo Chen, Zhuoer Feng, Zijun Yao, Ziwei Chai, Ziyuan Wang, Zuzhou Zhang, Bin Xu, Minlie Huang, Hongning Wang, Juanzi Li, Yuxiao Dong, and Jie Tang. Glm-5: from vibe coding to agentic engineering, 2026. URL https://arxiv.org/abs/2602.15763.

Daniel Goldstein, Eric Alcaide, Janna Lu, and Eugene Cheah. Radlads: Rapid attention distillation to linear attention decoders at scale, 2026. URL https://arxiv.org/abs/2505.03005.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

Yuxian Gu, Qinghao Hu, Shang Yang, Haocheng Xi, Junyu Chen, Song Han, and Han Cai. Jet-nemotron: Efficient language model with post neural architecture search. *arXiv preprint arXiv:2508.15884*, 2025.

Ali Hatamizadeh, Yejin Choi, and Jan Kautz. Gated deltanet-2: Decoupling erase and write in linear attention, 2026. URL https://arxiv.org/abs/2605.22791.

Stefan Heimersheim and Neel Nanda. How to use and interpret activation patching, 2024. URL https://arxiv.org/abs/2404.15255.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding, 2021. URL https://arxiv.org/abs/2009.03300.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. Ruler: What's the real context size of your long-context language models?, 2024. URL https://arxiv.org/abs/2404.06654.

Shengding Hu, Yuge Tu, Xu Han, Chaoqun He, Ganqu Cui, Xiang Long, Zhi Zheng, Yewei Fang, Yuxiang Huang, Weilin Zhao, Xinrong Zhang, Zheng Leng Thai, Kaihuo Zhang, Chongyi Wang, Yuan Yao, Chenyang Zhao, Jie Zhou, Jie Cai, Zhongwu Zhai, Ning Ding, Chao Jia, Guoyang Zeng, Dahai Li, Zhiyuan Liu, and Maosong Sun. Minicpm: Unveiling the potential of small language models with scalable training strategies, 2024. URL https://arxiv.org/abs/2404.06395.

Jungo Kasai, Hao Peng, Yizhe Zhang, Dani Yogatama, Gabriel Ilharco, Nikolaos Pappas, Yi Mao, Weizhu Chen, and Noah A. Smith. Finetuning pretrained transformers into rnns, 2021. URL https://arxiv.org/abs/2103.13076.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention, 2020. URL https://arxiv.org/abs/2006.16236.

Aakash Lahoti, Kevin Y. Li, Berlin Chen, Caitlin Wang, Aviv Bick, J. Zico Kolter, Tri Dao, and Albert Gu. Mamba-3: Improved sequence modeling using state space principles, 2026. URL https://arxiv.org/abs/2603.15569.

Disen Lan, Weigao Sun, Jiaxi Hu, Jusen Du, and Yu Cheng. Liger: Linearizing large language models to gated recurrent structures. In *Proceedings of the International Conference on Machine Learning (ICML)*. PMLR, 2025.

Yanhong Li, Songlin Yang, Shawn Tan, Mayank Mishra, Rameswar Panda, Jiawei Zhou, and Yoon Kim. Distilling to hybrid attention models via kl-guided layer selection, 2025. URL https://arxiv.org/abs/2512.20569.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt, 2023. URL https://arxiv.org/abs/2202.05262.

Weikang Meng, Liangyu Huo, Yadan Luo, Jiawen Guan, Jingyi Zhang, Yingjian Li, and Zheng Zhang. Still: Selecting tokens for intra-layer hybrid attention to linearize llms, 2026a. URL https://arxiv.org/abs/2602.02180.

Weikang Meng, Yadan Luo, Liangyu Huo, Yingjian Li, Yaowei Wang, Xin Li, and Zheng Zhang. Norm×direction: Restoring the missing query norm in vision linear attention, 2026b. URL https://arxiv.org/abs/2506.21137.

William Merrill, Yanhong Li, Tyler Romero, Anej Svete, Caia Costello, Pradeep Dasigi, Dirk Groeneveld, David Heineman, Bailey Kuehl, Nathan Lambert, Chuan Li, Kyle Lo, Saumya Malik, DJ Matusz, Benjamin Minixhofer, Jacob Morrison, Luca Soldaini, Finbarr Timbers, Pete Walsh, Noah A. Smith, Hannaneh Hajishirzi, and Ashish Sabharwal. Olmo hybrid: From theory to practice and back, 2026. URL https://arxiv.org/abs/2604.03444.

Paul Michel, Omer Levy, and Graham Neubig. Are sixteen heads really better than one?, 2019. URL https://arxiv.org/abs/1905.10650.

Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, and Hao Wu. Mixed precision training, 2018. URL https://arxiv.org/abs/1710.03740.

MiniMax, :, Aili Chen, Aonian Li, Bangwei Gong, Binyang Jiang, Bo Fei, Bo Yang, Boji Shan, Changqing Yu, Chao Wang, Cheng Zhu, Chengjun Xiao, Chengyu Du, Chi Zhang, Chu Qiao, Chunhao Zhang, Chunhui Du, Congchao Guo, Da Chen, Deming Ding, Dianjun Sun, Dong Li, Enwei Jiao, Haigang Zhou, Haimo Zhang, Han Ding, Haohai Sun, Haoyu Feng, Huaiguang Cai, Haichao Zhu, Jian Sun, Jiaqi Zhuang, Jiaren Cai, Jiayuan Song, Jin Zhu, Jingyang Li, Jinhao Tian, Jinli Liu, Junhao Xu, Junjie Yan, Junteng Liu, Junxian He, Kaiyi Feng, Ke Yang, Kecheng Xiao, Le Han, Leyang Wang, Lianfei Yu, Liheng Feng, Lin Li, Lin Zheng, Linge Du, Lingyu Yang, Lunbin Zeng, Minghui Yu, Mingliang Tao, Mingyuan Chi, Mozhi Zhang, Mujie Lin, Nan Hu, Nongyu Di, Peng Gao, Pengfei Li, Pengyu Zhao, Qibing Ren, Qidi Xu, Qile Li, Qin Wang, Rong Tian, Ruitao Leng, Shaoxiang Chen, Shaoyu Chen, Shengmin Shi, Shitong Weng, Shuchang Guan, Shuqi Yu, Sichen Li, Songquan Zhu, Tengfei Li, Tianchi Cai, Tianrun Liang, Weiyu Cheng, Weize Kong, Wenkai Li, Xiancai Chen, Xiangjun Song, Xiao Luo, Xiao Su, Xiaobo Li, Xiaodong Han, Xinzhu Hou, Xuan Lu, Xun Zou, Xuyang Shen, Yan Gong, Yan Ma, Yang Wang, Yiqi Shi, Yiran Zhong, Yonghong Duan, Yongxiang Fu, Yongyi Hu, Yu Gao, Yuanxiang Fan, Yufeng Yang, Yuhao Li, Yulin Hu, Yunan Huang, Yunji Li, Yunzhi Xu, Yuxin Mao, Yuxuan Shi, Yuze Wenren, Zehan Li, Zelin Li, Zhanxu Tian, Zhengmao Zhu, Zhenhua Fan, Zhenzhen Wu, Zhichao Xu, Zhihang Yu, Zhiheng Lyu, Zhuo Jiang, Zibo Gao, Zijia Wu, Zijian Song, and Zijun Sun. Minimax-m1: Scaling test-time compute efficiently with lightning attention, 2025. URL https://arxiv.org/abs/2506.13585.

MiniMax, :, Aili Chen, Aonian Li, Baichuan Zhou, Bangwei Gong, Binyang Jiang, Boji Dan, Changqing Yu, Chao Wang, Cheng Ma, Cheng Zhong, Cheng Zhu, Chengjun Xiao, Chengyi Yang, Chengyu Du, Chenyang Zhang, Chi Zhang, Chuangyi Huang, Chunhao Zhang, Chunhui Du, Chunyu Zhao, Congchao Guo, Da Chen, Deming Ding, Dianjun Sun, Dongyu Zhang, Enhui Yang, Fei Yu, Guang Zheng, Guodong Zheng, Guohong Li, Haichao Zhu, Haigang Zhou, Haimo Zhang, Han Ding, Hao Zhang, Haohai Sun, Haolin Lyu, Haonan Lu, Haoyu Wang, Huajie Shi, Huiyang Li, Jiacheng Chen, Jian Zhang, Jiaqi Zhuang, Jiaren Cai, Jiaxin Pan, Jiayao Li, Jiayuan Song, Jichuan Zhang, Jie Wang, Jihao Gu, Jin Zhu, Jingwei Dong, Jingyang Li, Jingyu Zhang, Jingze Zhuang, Jinhao Tian, Jinli Liu, Jinyi Hu, Jun Tao, Jun Zhang, Junbin Ruan, Junhao Xu, Junjie Yan, Junteng Liu, Junxian He, Kang Xu, Ke Ji, Ke Yang, Kecheng Xiao, Keyu Duan, Keyu Li, Le Han, Letian Ruan, Li Yuan, Lianfei Yu, Liheng Feng, Lijie Mo, Lin Li, Lingye Bao, Lingyu Yang, Lingyuan Zhou, Loki, Lu Chen, Lunbin Ceng, Ming Li, Ming Zhong, Mingliang Tao, Mingyuan Chi, Mujie Lin, Nan Hu, Ningxin Chen, Peiyin Zhu, Peng Gao, Pengcheng Gao, Pengfei Li, Penglin Li, Pengyu Zhao, Qibin Ren, Qidi Xu, Qihan Ren, Qile Li, Qin Wang, Quanliang Chen, Qunhong Ceng, Rong Tian, Rui Dong, Ruitao Leng, Ruize Zhang, Shanqi Liu, Shaoyu Chen, Sheng Jia, Shun Yao, Shuoran Zhao, Shuqi Yu, Sichen Li, Sicheng Pan, Songquan Zhu, Tengfei Li, Tian Xie, Tiancheng Qin, Tianrun Liang, Wei Liu, Weiqi Xu, Weitao Li, Weixiang Chen, Weiyu Cheng, Weiyu Zhang, Wenhu Chen, Wenqian Zhao, Xiancai Chen, Xiangjun Song, Xiangyuan Wang, Xiao Luo, Xiao Su, Xiaobo Li, Xiaodong Han, Xiaojie Wu, Xihao Song, Xingyi Han, Xinyu Guan, Xuan Lu, Xun Zou, Xunhao Lai, Xutong Li, Yan Gong, Yang Wang, Yang Xu, Yangsen Wang, Ye Tang, Yicheng Chen, Yinran Qiu, Yiqi Shi, Yiting Guo, Yiwen Huang, Yixuan Wang, Yongyi Hu, Yu Gao, Yu Zhang, Yuanxiang Ying, Yuanzhen Zhang, Yubo Wang, Yuchen Song, Yufeng Yang, Yuhang Meng, Yuhang Miao, Yuhao Li, Yujie Liu, Yulin Hu, Yunan Huang, Yunji Li, Yunyi Huang, Yusen Zhang, Yusu Hong, Yutao Xie, Yutong Zhang, Yuwen Liao, Yuxuan Shi, Yuze Wenren, Zebin Li, Zehan Li, Zejian Luo, Zeyu Jin, Zeyuan Sun, Zhanpeng Zhou, Zhaochen Su, Zhendong Li, Zhengmao Zhu, Zhengyuan Peng, Zhenhua Fan, Zhi Zhang, Zhichao Xu, Zhiheng Lv, Zhikang Xu, Zhitao He, Zhiwei He, Zhongyuan Li, Zibo Gao, Zijia Wu, Zijian Song, Zijian Zhou, Zijun Sun,

Zishan Huang, Ziying Chen, and Ziyue Ge. The minimax-m2 series: Mini activations unleashing max real-world intelligence, 2026. URL https://arxiv.org/abs/2605.26494.

Abhinav Moudgil, Ningyuan Huang, Eeshan Gunesh Dhekane, Pau Rodríguez, Luca Zappella, and Federico Danieli. Attention to mamba: A recipe for cross-architecture distillation, 2026. URL https://arxiv.org/abs/2604.14191.

NVIDIA, :, Aakshita Chandiramani, Aaron Blakeman, Abdullahi Olaoye, Abhibha Gupta, Abhilash Somasamudramath, Abhinav Khattar, Adeola Adesoba, Adi Renduchintala, Adil Asif, Aditya Agrawal, Aditya Vavre, Ahmad Kiswani, Aishwarya Padmakumar, Ajay Hotchandani, Akanksha Shukla, Akhiad Bercovich, Aleksander Ficek, Aleksandr Shaposhnikov, Alex Gronskiy, Alex Kondratenko, Alex Neefus, Alex Steiner, Alex Yang, Alexander Bukharin, Alexander Young, Ali Hatamizadeh, Ali Taghibakhshi, Alina Galiautdinova, Alisa Liu, Alok Kumar, Ameya Sunil Mahabaleshwarkar, Amir Klein, Amit Zuker, Amnon Geifman, Anahita Bhiwandiwalla, Ananth Subramaniam, Andrew Tao, Anjaney Shrivastava, Anjulie Agrusa, Ankur Srivastava, Ankur Verma, Ann Guan, Anna Shors, Annamalai Chockalingam, Anubhav Mandarwal, Aparnaa Ramani, Arham Mehta, Arti Jain, Arun Venkatesan, Asha Anoosheh, Ashwath Aithal, Ashwin Poojary, Asif Ahamed, Asit Mishra, Asli Sabanci Demiroz, Asma Kuriparambil Thekkumpate, Atefeh Sohrabizadeh, Avinash Kaur, Ayush Dattagupta, Barath Subramaniam Anandan, Bardiya Sadeghi, Barnaby Simkin, Ben Lanir, Benedikt Schifferer, Benjamin Chislett, Besmira Nushi, Bilal Kartal, Bill Thiede, Bita Darvish Rouhani, Bobby Chen, Boris Ginsburg, Brandon Norick, Branislav Kisacanin, Brian Yu, Bryan Catanzaro, Buvaneswari Mani, Carlo del Mundo, Chankyu Lee, Chanran Kim, Chantal Hwang, Chao Ni, Charles Wang, Charlie Truong, Cheng-Ping Hsieh, Chenhan Yu, Chenjie Luo, Cherie Wang, Chetan Mungekar, Chintan Patel, Chris Alexiuk, Chris Holguin, Chris Wing, Christian Munley, Christopher Parisien, Chuck Desai, Chunyang Sheng, Collin Neale, Cyril Meurillon, Dakshi Kumar, Dan Gil, Dan Su, Dane Corneil, Daniel Afrimi, Daniel Burkhardt Eliuth Triana, Daniel Egert, Daniel Fatade, Daniel Lo, Daniel Rohrer, Daniel Serebrenik, Daniil Sorokin, Daria Gitman, Daria Levy, Darko Stosic, David Edelsohn, David Messina, David Mosallanezhad, David Tamok, Deena Donia, Deepak Narayanan, Devin O'Kelly, Dheeraj Peri, Dhruv Nathawani, Di Wu, Dima Rekesh, Dina Yared, Divyanshu Kakwani, Dmitry Konyagin Brandon Tuttle, Dong Ahn, Dongfu Jiang, Dorrin Poorkay, Douglas O'Flaherty, Duncan Riach, Dusan Stosic, Dustin Van Stee, Edgar Minasyan, Edward Lin, Eileen Peters Long, Elad Segal, Elena Lantz, Elena Lewis, Ellie Evans, Elliott Ning, Eric Chung, Eric Harper, Eric Pham-Hung, Eric W. Tramel, Erick Galinkin, Erik Pounds, Esti Etrog, Evan Briones, Evan Wu, Evelina Bakhturina, Evgeny Tsykunov, Ewa Dobrowolska, Farshad Saberi Movahed, Farzan Memarian, Fay Wang, Fei Jia, Felipe Soares, Felipe Vieira Frujeri, Feng Chen, Fengguang Lin, Ferenc Galko, Fortuna Zhang, Frankie Siino, Frida Hou, Gantavya Bhatt, Gargi Prasad, Geethapriya Venkataramani, Geetika Gupta, George Armstrong, Gerald Shen, Giulio Borghesi, Gordana Neskovic, Gorkem Batmaz, Grace Lam, Grace Wu, Greg Pauloski, Greyson Davis, Grigor Nalbandyan, Guoming Zhang, Guy Farber, Guyue Huang, Haifeng Qian, Haran Kumar Shiv Kumar, Harry Kim, Harsh Sharma, Hayate Iso, Hayley Ross, Herbert Hum, Herman Sahota, Hexin Wang, Himanshu Soni, Hiren Upadhyay, Huy Nguyen, Iain Cunningham, Ido Galil, Ido Shahaf, Igino Padovani, Igor Gitman, Igor Shovkun, Ikroop Dhillon, Ilya Loshchilov, Ingrid Kelly, Itamar Schen, Itay Levy, Ivan Moshkov, Izik Golan, Izzy Putterman, Jain Tu, Jan Baczek, Jan Kautz, Jane Polak Scowcroft, Janica Rosenberg, Jared Casper, Jarrod Pflum, Jason Grant, Jason Sewall, Jatin Mitra, Jeffrey Glick, Jenny Chen, Jesse Oliver, Jiacheng Xu, Jiafan Zhu, Jialin Song, Jian Zhang, Jiaqi Zeng, Jie Lou, Jill Milton, Jim Chow, Jimmy Zhang, Jinhang Choi, Jining Huang, Jocelyn Huang, Joel Caruso, Joey Conway, Joey Guman, Johan Jatko, John Kamalu, Johnny Greco, Jonathan Cohen, Jonathan Raiman, Joseph Jennings, Joyjit Daw, Juan Yu, Julio Tapia, Junkeun Yi, Jupinder Parmar, Jyothi Achar, Kari Briski, Kartik Mattoo, Katherine Cheung, Katherine Luna, Keith Wyss, Kevin Shih, Kezhi Kong, Khanh Nguyen, Khushi Bhardwaj, Kirill Buryak, Kirthi Shankar Sivamani, Konstantinos Krommydas, Kris Murphy, Krishna C. Puvvada, Krzysztof Pawelec, Kumar Anik, Laikh Tewari, Laya Sleiman, Leo Du, Leon Derczynski, Li Ding, Lilach Ilan, Lingjie Wu, Lizzie Wei, Luis Vega, Lun Su, Maarten Van Segbroeck, Maer Rodrigues de Melo, Magaret Zhang, Mahan Fathi, Makesh Narsimhan Sreedhar, Makesh Sreedhar, Makesh Tarun Chandran, Manuel Reyes Gomez, Maor Ashkenazi, Marc Cuevas, Marc Romeijn, Margaret Zhang, Mark Cai, Mark Gabel, Markus Kliegl, Martyna Patelka, Maryam Moosaei, Matthew Varacalli, Matvei Novikov, Mauricio Ferrato, Mehrzad Samadi, Melissa Corpuz, Meng Xin, Mengdi Wang, Mengru Wang, Meredith Price, Micah Schaffer, Michael Andersch, Michael Boone, Michael Evans, Michael Z Wang, Miguel Martinez, Mikail Khona, Mike Chrzanowski, Mike Hollinger, Mingyuan Ma, Minseok Lee, Mohammad Dabbah, Mohammad Shoeybi, Mostofa

Patwary, Nabin Mulepati, Nader Khalil, Najeeb Nabwani, Nancy Agarwal, Nanthini Balasubramaniam, Narimane Hennouni, Narsi Kodukula, Natalie Hereth, Nathaniel Pinckney, Nave Assaf, Negar Habibi, Nestor Qin, Neta Zmora, Netanel Haber, Nick Reamaroon, Nickson Quak, Nidhi Bhatia, Nikhil Jukar, Nikki Pope, Nikolai Ludwig, Nima Tajbakhsh, Nir Ailon, Nirmal Juluru, Nirmalya De, Nowel Pitt, Oleg Rybakov, Oleksii Hrinchuk, Oleksii Kuchaiev, Olivier Delalleau, Oluwatobi Olabiyi, Omer Ullman Argov, Omri Almog, Omri Puny, Oren Tropp, Otavio Padovani, Ouye Xie, Parth Chadha, Pasha Shamis, Paul Gibbons, Pavlo Molchanov, Peter Belcak, Peter Jin, Pinky Xu, Piotr Januszewski, Pooya Jannaty, Prachi Shevate, Pradeep Thalasta, Pranav Prashant Thombre, Prasoon Varshney, Prerana Gambhir, Pritam Gundecha, Przemek Tredak, Qing Miao, Qiyu Wan, Quan Tran Minh, Rabeeh Karimi Mahabadi, Rachel Oberman, Rachit Garg, Rahul Kandu, Raina Zhong, Ran El-Yaniv, Ran Zilberstein, Rasoul Shafipour, Renee Yao, Renjie Pi, Richard Mazzarese, Richard Wang, Rick Izzo, Ridhima Singla, Rima Shahbazyan, Rishabh Garg, Ritika Borkar, Ritu Gala, Riyad Islam, Robert Clark, Robert Hesse, Roger Waleffe, Rohit Varma Kalidindi, Rohit Watve, Roi Koren, Ron Fan, Ruchika Kharwar, Ruisi Cai, Ruoxi Zhang, Russell J. Hewett, Ryan Prenger, Ryan Timbrook, Ryota Egashira, Sadegh Mahdavi, Sagar Singh Ashutosh Joshi, Sahil Modi, Samuel Kriman, Sandeep Pombra, Sanjay Kariyappa, Sanjeev Satheesh, Santiago Pombo, Saori Kaji, Satish Pasumarthi, Saurav Mishra, Saurav Muralidharan, Scott Hara, Sean Narenthiran, Sebastian Rogawski, Seonjin Na, Seonmyeong Bak, Sepehr Sameni, Seth Poulos, Shahar Mor, Shantanu Acharya, Shaona Ghosh Adam Lord, Sharath Turuvekere Sreenivas, Shaun Kotek, Shaya Gharghabi, Shelby Thomas, Sheng-Chieh Lin, Shibani Likhite, Shiqing Fan, Shiyang Chen, Shreya Gopal, Shrimai Prabhumoye, Shubham Pachori, Shubham Toshniwal, Shuo Zhang, Shuoyang Ding, Shyam Renjith, Shyamala Prayaga, Siddhartha Jain, Simeng Sun, Sirisha Rella, Sirshak Das, Smita Ithape, Sneha Harishchandra S, Somshubra Majumdar, Soumye Singhal, Sri Harsha Singudasu, Sriharsha Niverty, Stas Sergienko, Stefana Gloginic, Stefania Alborghetti, Stephen Ge, Stephen McCullough, Sugam Dipak Devare, Suguna Varshini Velury, Sukrit Rao, Sumeet Kumar Barua, Sunny Gai, Suseella Panguluri, Sushil Koundinyan, Swathi Patnam, Sweta Priyadarshi, Swetha Bhendigeri, Syeda Nahida Akter, Sylendran Arunagiri, Tailling Yuan, Talor Abramovich, Tan Bui, Tan Yu, Terry Kong, Thanh Do, Thomas Gburek, Thorgane Marques, Tiffany Moore, Tijmen Blankevoort, Tim Moon, Timothy Ma, Tiyasa Mitra, Tomasz Grzegorzek, Tomer Asida, Tomer Bar Natan, Tomer Keren, Tomer Ronen, Traian Rebedea, Trenton Starkey, Tugrul Konuk, Twinkle Vashishth, Tyler Condensa, Udi Karpas, Ushnish De, Vahid Noorozi, Vahid Noroozi, Vanshil Atul Shah, Veena Vaidyanathan, Venkat Srinivasan, Venmugil Elango, Victor Cui, Vijay Korthikanti, Vikas Mehta, Virginia Adams, Virginia Wu, Vitaly Kurin, Vitaly Lavrukhin, Vladimir Anisimov, Wan Seo, Wanli Jiang, Wasi Uddin Ahmad, Wei Du, Wei Ping, Wei-Ming Chen, Wendy Quan, Wenliang Dai, Wenwen Gao, Will Jennings, William Zhang, Xiaowei Ren, Xiaowen Xin, Xin Li, Yang Yu, Yangyi Chen, Yaniv Galron, Yashaswi Karnati, Yejin Choi, Yev Meyer, Yi-Fu Wu, Yian Zhang, Ying Lin, Yonatan Geifman, Yonggan Fu, Yoshi Suhara, Youngeun Kwon, Yuan Zhang, Yuki Huang, Zach Moshe, Zhilin Wang, Zhiyu Cheng, Zhongbo Zhu, Zhuolin Yang, Zihan Liu, Zijia Chen, Zijie Yan, and Zuhair Ahmed. Nemotron 3 super: Open, efficient mixture-of-experts hybrid mamba-transformer model for agentic reasoning, 2026. URL https://arxiv.org/abs/2604.12374.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Quan Ngoc Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. The lambada dataset: Word prediction requiring a broad discourse context, 2016. URL https://arxiv.org/abs/1606.06031.

Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. The fineweb datasets: Decanting the web for the finest text data at scale, 2024. URL https://arxiv.org/abs/2406.17557.

Zhen Qin, Weigao Sun, Dong Li, Xuyang Shen, Weixuan Sun, and Yiran Zhong. Various lengths, constant speed: Efficient language modeling with lightning attention, 2024. URL https://arxiv.org/abs/2405.17381.

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.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *arXiv preprint arXiv:1907.10641*, 2019.

Imanol Schlag, Kazuki Irie, and Jürgen Schmidhuber. Linear transformers are secretly fast weight programmers, 2021. URL https://arxiv.org/abs/2102.11174.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V. Le, Ed H. Chi, Denny Zhou, and Jason Wei. Challenging big-bench tasks and whether chain-of-thought can solve them, 2022. URL https://arxiv.org/abs/2210.09261.

Zecheng Tang, Quantong Qiu, Yi Yang, Zhiyi Hong, Haiya Xiang, Kebin Liu, Qingqing Dang, Juntao Li, and Min Zhang. Elastic attention: Test-time adaptive sparsity ratios for efficient transformers, 2026. URL https://arxiv.org/abs/2601.17367.

Core Team, Bangjun Xiao, Bingquan Xia, Bo Yang, Bofei Gao, Bowen Shen, Chen Zhang, Chenhong He, Chiheng Lou, Fuli Luo, Gang Wang, Gang Xie, Hailin Zhang, Hanglong Lv, Hanyu Li, Heyu Chen, Hongshen Xu, Houbin Zhang, Huaqiu Liu, Jiangshan Duo, Jianyu Wei, Jiebao Xiao, Jinhao Dong, Jun Shi, Junhao Hu, Kainan Bao, Kang Zhou, Lei Li, Liang Zhao, Linghao Zhang, Peidian Li, Qianli Chen, Shaohui Liu, Shihua Yu, Shijie Cao, Shimao Chen, Shouqiu Yu, Shuo Liu, Tianling Zhou, Weijiang Su, Weikun Wang, Wenhan Ma, Xiangwei Deng, Bohan Mao, Bowen Ye, Can Cai, Chenghua Wang, Chengxuan Zhu, Chong Ma, Chun Chen, Chunan Li, Dawei Zhu, Deshan Xiao, Dong Zhang, Duo Zhang, Fangyue Liu, Feiyu Yang, Fengyuan Shi, Guoan Wang, Hao Tian, Hao Wu, Heng Qu, Hongfei Yi, Hongxu An, Hongyi Guan, Xing Zhang, Yifan Song, Yihan Yan, Yihao Zhao, Yingchun Lai, Yizhao Gao, Yu Cheng, Yuanyuan Tian, Yudong Wang, Zhen Tang, Zhengju Tang, Zhengtao Wen, Zhichao Song, Zhixian Zheng, Zihan Jiang, Jian Wen, Jiarui Sun, Jiawei Li, Jinlong Xue, Jun Xia, Kai Fang, Menghang Zhu, Nuo Chen, Qian Tu, Qihao Zhang, Qiying Wang, Rang Li, Rui Ma, Shaolei Zhang, Shengfan Wang, Shicheng Li, Shuhao Gu, Shuhuai Ren, Sirui Deng, Tao Guo, Tianyang Lu, Weiji Zhuang, Weikang Zhang, Weimin Xiong, Wenshan Huang, Wenyu Yang, Xin Zhang, Xing Yong, Xu Wang, Xueyang Xie, Yilin Jiang, Yixin Yang, Yongzhe He, Yu Tu, Yuanliang Dong, Yuchen Liu, Yue Ma, Yue Yu, Yuxing Xiang, Zhaojun Huang, Zhenru Lin, Zhipeng Xu, Zhiyang Chen, Zhonghua Deng, Zihan Zhang, and Zihao Yue. Mimo-v2-flash technical report, 2026a. URL https://arxiv.org/abs/2601.02780.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, Gaël Liu, Francesco Visin, Kathleen Kenealy, Lucas Beyer, Xiaohai Zhai, Anton Tsitsulin, Robert Busa-Fekete, Alex Feng, Noveen Sachdeva, Benjamin Coleman, Yi Gao, Basil Mustafa, Iain Barr, Emilio Parisotto, David Tian, Matan Eyal, Colin Cherry, Jan-Thorsten Peter, Danila Sinopalnikov, Surya Bhupatiraju, Rishabh Agarwal, Mehran Kazemi, Dan Malkin, Ravin Kumar, David Vilar, Idan Brusilovsky, Jiaming Luo, Andreas Steiner, Abe Friesen, Abhanshu Sharma, Abheesht Sharma, Adi Mayrav Gilady, Adrian Goedeckemeyer, Alaa Saade, Alex Feng, Alexander Kolesnikov, Alexei Bendebury, Alvin Abdagic, Amit Vadi, András György, André Susano Pinto, Anil Das, Ankur Bapna, Antoine Miech, Antoine Yang, Antonia Paterson, Ashish Shenoy, Ayan Chakrabarti, Bilal Piot, Bo Wu, Bobak Shahriari, Bryce Petrini, Charlie Chen, Charline Le Lan, Christopher A. Choquette-Choo, CJ Carey, Cormac Brick, Daniel Deutsch, Danielle Eisenbud, Dee Cattle, Derek Cheng, Dimitris Paparas, Divyashree Shivakumar Sreepathihalli, Doug Reid, Dustin Tran, Dustin Zelle, Eric Noland, Erwin Huizenga, Eugene Kharitonov, Frederick Liu, Gagik Amirkhanyan, Glenn Cameron, Hadi Hashemi, Hanna Klimczak-Plucińska, Harman Singh, Harsh Mehta, Harshal Tushar Lehri, Hussein Hazimeh, Ian Ballantyne, Idan Szpektor, Ivan Nardini, Jean Pouget-Abadie, Jetha Chan, Joe Stanton, John Wieting, Jonathan Lai, Jordi Orbay, Joseph Fernandez, Josh Newlan, Ju yeong Ji, Jyotinder Singh, Kat Black, Kathy Yu, Kevin Hui, Kiran Vodrahalli, Klaus Greff, Linhai Qiu, Marcella Valentine, Marina Coelho, Marvin Ritter, Matt Hoffman, Matthew Watson, Mayank Chaturvedi, Michael Moynihan, Min Ma, Nabila Babar, Natasha Noy, Nathan Byrd, Nick Roy, Nikola Momchev, Nilay Chauhan, Noveen Sachdeva, Oskar Bunyan, Pankil Botarda, Paul Caron, Paul Kishan Rubenstein, Phil Culliton, Philipp Schmid, Pier Giuseppe Sessa, Pingmei Xu, Piotr Stanczyk, Pouya Tafti, Rakesh Shivanna, Renjie Wu, Renke Pan, Reza Rokni, Rob Willoughby, Rohith Vallu, Ryan Mullins, Sammy Jerome, Sara Smoot, Sertan Girgin, Shariq Iqbal, Shashir Reddy, Shruti Sheth, Siim Põder, Sijal Bhatnagar,

Sindhu Raghuram Panyam, Sivan Eiger, Susan Zhang, Tianqi Liu, Trevor Yacovone, Tyler Liechty, Uday Kalra, Utku Evci, Vedant Misra, Vincent Roseberry, Vlad Feinberg, Vlad Kolesnikov, Woohyun Han, Woosuk Kwon, Xi Chen, Yinlam Chow, Yuvein Zhu, Zichuan Wei, Zoltan Egyed, Victor Cotruta, Minh Giang, Phoebe Kirk, Anand Rao, Kat Black, Nabila Babar, Jessica Lo, Erica Moreira, Luiz Gustavo Martins, Omar Sanseviero, Lucas Gonzalez, Zach Gleicher, Tris Warkentin, Vahab Mirrokni, Evan Senter, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, Yossi Matias, D. Sculley, Slav Petrov, Noah Fiedel, Noam Shazeer, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Jean-Baptiste Alayrac, Rohan Anil, Dmitry, Lepikhin, Sebastian Borgeaud, Olivier Bachem, Armand Joulin, Alek Andreev, Cassidy Hardin, Robert Dadashi, and Léonard Hussenot. Gemma 3 technical report, 2025a. URL https://arxiv.org/abs/2503.19786.

Kimi Team, Yu Zhang, Zongyu Lin, Xingcheng Yao, Jiaxi Hu, Fanqing Meng, Chengyin Liu, Xin Men, Songlin Yang, Zhiyuan Li, Wentao Li, Enzhe Lu, Weizhou Liu, Yanru Chen, Weixin Xu, Longhui Yu, Yejie Wang, Yu Fan, Longguang Zhong, Enming Yuan, Dehao Zhang, Yizhi Zhang, T. Y. Liu, Haiming Wang, Shengjun Fang, Weiran He, Shaowei Liu, Yiwei Li, Jianlin Su, Jiezhong Qiu, Bo Pang, Junjie Yan, Zhejun Jiang, Weixiao Huang, Bohong Yin, Jiacheng You, Chu Wei, Zhengtao Wang, Chao Hong, Yutian Chen, Guanduo Chen, Yucheng Wang, Huabin Zheng, Feng Wang, Yibo Liu, Mengnan Dong, Zheng Zhang, Siyuan Pan, Wenhao Wu, Yuhao Wu, Longyu Guan, Jiawen Tao, Guohong Fu, Xinran Xu, Yuzhi Wang, Guokun Lai, Yuxin Wu, Xinyu Zhou, Zhilin Yang, and Yulun Du. Kimi linear: An expressive, efficient attention architecture, 2025b. URL https://arxiv.org/abs/2510.26692.

Kimi Team, Yifan Bai, Yiping Bao, Y. Charles, Cheng Chen, Guanduo Chen, Haiting Chen, Huarong Chen, Jiahao Chen, Ningxin Chen, Ruijue Chen, Yanru Chen, Yuankun Chen, Yutian Chen, Zhuofu Chen, Jialei Cui, Hao Ding, Mengnan Dong, Angang Du, Chenzhuang Du, Dikang Du, Yulun Du, Yu Fan, Yichen Feng, Kelin Fu, Bofei Gao, Chenxiao Gao, Hongcheng Gao, Peizhong Gao, Tong Gao, Yuyao Ge, Shangyi Geng, Qizheng Gu, Xinran Gu, Longyu Guan, Haiqing Guo, Jianhang Guo, Xiaoru Hao, Tianhong He, Weiran He, Wenyang He, Yunjia He, Chao Hong, Hao Hu, Yangyang Hu, Zhenxing Hu, Weixiao Huang, Zhiqi Huang, Zihao Huang, Tao Jiang, Zhejun Jiang, Xinyi Jin, Yongsheng Kang, Guokun Lai, Cheng Li, Fang Li, Haoyang Li, Ming Li, Wentao Li, Yang Li, Yanhao Li, Yiwei Li, Zhaowei Li, Zheming Li, Hongzhan Lin, Xiaohan Lin, Zongyu Lin, Chengyin Liu, Chenyu Liu, Hongzhang Liu, Jingyuan Liu, Junqi Liu, Liang Liu, Shaowei Liu, T. Y. Liu, Tianwei Liu, Weizhou Liu, Yangyang Liu, Yibo Liu, Yiping Liu, Yue Liu, Zhengying Liu, Enzhe Lu, Haoyu Lu, Lijun Lu, Yashuo Luo, Shengling Ma, Xinyu Ma, Yingwei Ma, Shaoguang Mao, Jie Mei, Xin Men, Yibo Miao, Siyuan Pan, Yebo Peng, Ruoyu Qin, Zeyu Qin, Bowen Qu, Zeyu Shang, Lidong Shi, Shengyuan Shi, Feifan Song, Jianlin Su, Zhengyuan Su, Lin Sui, Xinjie Sun, Flood Sung, Yunpeng Tai, Heyi Tang, Jiawen Tao, Qifeng Teng, Chaoran Tian, Chensi Wang, Dinglu Wang, Feng Wang, Hailong Wang, Haiming Wang, Jianzhou Wang, Jiaxing Wang, Jinhong Wang, Shengjie Wang, Shuyi Wang, Si Wang, Xinyuan Wang, Yao Wang, Yejie Wang, Yiqin Wang, Yuxin Wang, Yuzhi Wang, Zhaoji Wang, Zhengtao Wang, Zhengtao Wang, Zhexu Wang, Chu Wei, Qianqian Wei, Haoning Wu, Wenhao Wu, Xingzhe Wu, Yuxin Wu, Chenjun Xiao, Jin Xie, Xiaotong Xie, Weimin Xiong, Boyu Xu, Jinjing Xu, L. H. Xu, Lin Xu, Suting Xu, Weixin Xu, Xinran Xu, Yangchuan Xu, Ziyao Xu, Jing Xu, Jing Xu, Junjie Yan, Yuzi Yan, Hao Yang, Xiaofei Yang, Yi Yang, Ying Yang, Zhen Yang, Zhilin Yang, Zonghan Yang, Haotian Yao, Xingcheng Yao, Wenjie Ye, Zhuorui Ye, Bohong Yin, Longhui Yu, Enming Yuan, Hongbang Yuan, Mengjie Yuan, Siyu Yuan, Haobing Zhan, Dehao Zhang, Hao Zhang, Wanlu Zhang, Xiaobin Zhang, Yadong Zhang, Yangkun Zhang, Yichi Zhang, Yizhi Zhang, Yongting Zhang, Yu Zhang, Yutao Zhang, Yutong Zhang, Zheng Zhang, Haotian Zhao, Yikai Zhao, Zijia Zhao, Huabin Zheng, Shaojie Zheng, Longguang Zhong, Jianren Zhou, Xinyu Zhou, Zaida Zhou, Jinguo Zhu, Zhen Zhu, Weiyu Zhuang, and Xinxing Zu. Kimi k2: Open agentic intelligence, 2026b. URL https://arxiv.org/abs/2507.20534.

Qwen Team. Qwen3.5: Towards native multimodal agents. https://qwen.ai/blog, April 2026. URL https://qwen.ai/blog. Official Blog Post.

Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, et al. Scaling monosemanticity: Extracting interpretable features from Claude 3 Sonnet. *arXiv preprint arXiv:2605.29358*, 2026.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned, 2019. URL https://arxiv.org/abs/1905.09418.

Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in gpt-2 small, 2022. URL https://arxiv.org/abs/2211.00593.

Guangxuan Xiao, Jiaming Tang, Jingwei Zuo, Junxian Guo, Shang Yang, Haotian Tang, Yao Fu, and Song Han. Duoattention: Efficient long-context llm inference with retrieval and streaming heads, 2024a. URL https://arxiv.org/abs/2410.10819.

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

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025a. URL https://arxiv.org/abs/2505.09388.

Songlin Yang, Jan Kautz, and Ali Hatamizadeh. Gated delta networks: Improving mamba2 with delta rule. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL https://openreview.net/forum?id=r8H7xhYPwz.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019.

Michael Zhang, Simran Arora, Rahul Chalamala, Alan Wu, Benjamin Spector, Aaryan Singhal, Krithik Ramesh, and Christopher Ré. Lolcats: On low-rank linearizing of large language models, 2025. URL https://arxiv.org/abs/2410.10254.

Wei Zhang, Chaoqun Wan, Yonggang Zhang, Yiu ming Cheung, Xinmei Tian, Xu Shen, and Jieping Ye. Interpreting and improving large language models in arithmetic calculation, 2024. URL https://arxiv.org/abs/2409.01659.

Yusheng Zhao, Hourun Li, Bohan Wu, Yichun Yin, Lifeng Shang, Jingyang Yuan, Meng Zhang, and Ming Zhang. Switch attention: Towards dynamic and fine-grained hybrid transformers, 2026. URL https://arxiv.org/abs/2603.26380.

A Preliminaries

A.1 Causal Patching Methods

Activation patching. Given a clean input x and a counterfactual input x', activation patching replaces selected clean-run activations with the corresponding activations from x' and evaluates the target behavior (Heimersheim & Nanda, 2024). For answer prediction, a common scalar readout is the correct-versus-counterfactual logit difference:

$$m(x) = z[a^+] - z[a^-], \tag{5}$$

where z is the output-logit vector and a^+ and a^- identify the correct and counterfactual answer tokens. These token identities remain fixed when comparing executions for the same pair. A decrease measures the effect of the specified replacement on this readout. The receiver intervention used in HydraHead fixes downstream attention outputs to their clean values while recomputing downstream MLPs (Section 2.1).

Path patching. Path patching restricts an intervention to a selected computational pathway, allowing the contribution of an upstream head to be measured through a chosen downstream receiver (Wang et al., 2022). We use it to obtain sender scores and trace upstream from the receiver set. The head-scoring procedure is defined in Section 2.1; its retrieval instantiation is described in Appendix F.1.

A.2 Grouped-Query Attention Mechanism

Modern large language models predominantly adopt Grouped-Query Attention (GQA) (Ainslie et al., 2023) to balance inference efficiency and modeling capacity. Thus, we consider a Transformer model with L layers, where each layer consists of a GQA and a Multi-Layer Perceptron (MLP). Let H denote the total number of query heads and G denote the number of key-value heads, with $G < H$. The set of query heads $\mathcal{H} = \{1, \ldots, H\}$ is partitioned into G disjoint groups $\mathcal{G} = \{1, \ldots, G\}$, such that all query heads in a group share a single key-value head.

Given an input sequence represented as a matrix $\mathbf{X} \in \mathbb{R}^{T\times d}$, where T is the sequence length and d is the hidden dimension, we define the head dimension as $d_h = d/H$. In standard GQA implementations, the key and value heads share the same dimension d_h as the query heads.

The attention mechanism employs three global learnable projection matrices: $\mathbf{W}^Q \in \mathbb{R}^{d\times d}$, $\mathbf{W}^K \in \mathbb{R}^{d\times d_G}$, and $\mathbf{W}^V \in \mathbb{R}^{d\times d_G}$, where $d_G = d \cdot \frac{G}{H}$ is the total dimension for all key/value heads. The global projected representations are computed as:

$$\mathbf{Q} = \mathbf{X}\mathbf{W}^Q \in \mathbb{R}^{T\times d}, \quad \mathbf{K} = \mathbf{X}\mathbf{W}^K \in \mathbb{R}^{T\times d_G}, \quad \mathbf{V} = \mathbf{X}\mathbf{W}^V \in \mathbb{R}^{T\times d_G}. \tag{6}$$

These tensors are split along the feature dimension to obtain individual head representations. For each query head $h \in \mathcal{H}$, let $\mathbf{Q}_h \in \mathbb{R}^{T\times d_h}$ denote the slice of $\mathbf{Q}$ corresponding to the h-th head. Similarly, for each key-value group $g \in \mathcal{G}$, let $\mathbf{K}_g \in \mathbb{R}^{T\times d_h}$ and $\mathbf{V}_g \in \mathbb{R}^{T\times d_h}$ denote the slices of $\mathbf{K}$ and $\mathbf{V}$ corresponding to the g-th group. Note that each $\mathbf{K}_g$ and $\mathbf{V}_g$ has dimension d_h, matching the query head dimension.

Let $g(h) \in \mathcal{G}$ be the mapping function that assigns query head h to its corresponding key-value group. The shared key and value matrices for the group associated with head h are denoted as $\mathbf{K}_{g(h)}$ and $\mathbf{V}_{g(h)}$. In the standard FA mode, the output matrix for query head h is computed as:

$$\mathbf{O}_h = \text{softmax}\left(\frac{\mathbf{Q}_h\mathbf{K}_{g(h)}^\top}{\sqrt{d_h}}\right)\mathbf{V}_{g(h)}. \tag{7}$$

The outputs from all heads are concatenated along the feature dimension to form $\mathbf{O} = [\mathbf{O}_1, \mathbf{O}_2, ..., \mathbf{O}_H] \in \mathbb{R}^{T\times d}$, which is then projected to produce the final layer output $\mathbf{Y} \in \mathbb{R}^{T\times d}$:

$$\mathbf{Y} = \mathbf{O}\mathbf{W}^O, \tag{8}$$

where $\mathbf{W}^O \in \mathbb{R}^{d\times d}$ is the output projection matrix.

A.3 Linear Attention with Gated DeltaNet (GDN)

Although GQA reduces the KV cache footprint, its quadratic computational complexity still hinders the deployment of models in long-context scenarios, thereby motivating the development of linear attention mechanisms. Given that GDN (Yang et al., 2025b) has already been adopted in industrial-grade hybrid models, we likewise select GDN as the representative linear attention layer.

Standard Linear Attention. Standard linear attention (Katharopoulos et al., 2020) maintains a matrix-valued recurrent state $\mathbf{S}_t \in \mathbb{R}^{d_h\times d_h}$ that accumulates key-value associations. Given head-wise input projections $\mathbf{q}_t, \mathbf{k}_t, \mathbf{v}_t \in \mathbb{R}^{d_h}$ at step t, the state update and output are defined as:

$$\mathbf{S}_t = \mathbf{S}_{t-1} + \mathbf{k}_t\mathbf{v}_t^\top, \tag{9}$$

$$\mathbf{y}_t = \mathbf{S}_t^\top\mathbf{q}_t. \tag{10}$$

While computationally efficient, this formulation lacks a forgetting mechanism, causing the state to grow unbounded and leading to significant interference over long contexts.

Gated DeltaNet. To address these limitations, GDN interprets the state update as online gradient descent on a reconstruction loss, incorporating both a delta rule and a gating mechanism. Specifically, GDN introduces a scalar forget gate $\alpha_t \in (0, 1]$ and a learning rate β_t to regulate memory retention. The state update rule is formulated as:

$$\mathbf{S}_t = \alpha_t \left(\mathbf{I} - \beta_t \mathbf{k}_t \mathbf{k}_t^\top\right) \mathbf{S}_{t-1} + \beta_t \mathbf{k}_t \mathbf{v}_t^\top, \tag{11}$$

where $\mathbf{I} \in \mathbb{R}^{d_h \times d_h}$ is the identity matrix. The term $\left(\mathbf{I} - \beta_t \mathbf{k}_t \mathbf{k}_t^\top\right)$ acts as a projection that corrects the state based on the current key, while α_t serves as a data-dependent weight decay factor, controlling the lifespan of stored memories. The output is then computed as:

$$\mathbf{y}_t = \mathbf{S}_t^\top \mathbf{q}_t. \tag{12}$$

This formulation allows GDN to maintain a constant-size hidden state $\mathbf{S}$ while dynamically balancing the retention of new information and the forgetting of outdated associations, thereby facilitating stable and efficient linear-time inference.

B More Details of Our Method

B.1 Details of Branch-specific Refinement

The internal structural designs of the two attention types—such as positional encoding, gating mechanisms, and key-value head expansion—also exert a significant influence on head-wise hybridization. While HypeNet (Chen et al., 2026) validates the effectiveness of these designs in the context of layer-wise mixing, we verify them under the setting of head-wise hybridization (Figure 4).

For the FA branch, we aim to enhance attention stability in long-context, especially extended-context scenarios. To this end, we depart from the standard Rotary Position Embedding (RoPE) protocol: following HypeNet (Chen et al., 2026) and DroPE (Gelberg et al., 2025), we remove RoPE and instead apply a log-scale coefficient to the query features after training, which effectively stabilizes the attention distribution. In addition, motivated by gated attention (Qiu et al., 2025), we introduce an auxiliary gate branch within the FA module. This design not only boosts the model's representational capacity but also effectively alleviates the "attention sink" phenomenon commonly observed in high-precision attention heads (Xiao et al., 2024b).

For the GDN branch, we aim to strengthen its positional awareness within the inherent receptive field. Retaining the native short convolution and gating mechanism for local pattern recognition and stable state transitions, we explicitly integrate RoPE into the query and key projections to compensate for the limited positional sensitivity of linear recurrence, enabling precise capture of relative positional relationships within the perception window. Furthermore, to enhance representational capacity, we expand the number of key-value heads to match that of query heads—effectively transitioning from a GQA-like to an MHA-like configuration. Although described here in the context of GDN, these enhancements are directly transferable to other linear attention variants.

B.2 Details of Hybrid Transfer Learning

Here, we provide the implementation details of the three-stage transfer pipeline outlined in Section 2.3.

B.2.1 Stage 1: Parameter Migration and Layer-wise Output Alignment

The first stage focuses on initializing the hybrid architecture and aligning the local behavior of each layer. After identifying the critical heads to remain as FA via our interpretability-driven selection mechanism, we initialize the remaining heads as GDN branches.

Parameter Migration Initialization. We employ a careful weight initialization strategy to minimize the distribution shift at the beginning of training:

- **FA Branch with Gate Modulation:** For the heads retained as FA, we introduce a lightweight gate branch to allow for dynamic modulation during later stages. To encourage the gate to initially approximate an identity function (i.e., maintaining fidelity to the original FA output), we initialize

the gate branch projection weights to near-zero and set the bias so that initial gate values are close to 1. This guarantees that the FA branch starts with its pre-trained functionality intact, providing a stable anchor for the hybrid model.

- **GDN Branch Weight Reuse:** For the heads converted to GDN, we directly reuse the Q, K, V projection weights from the original pre-trained FA layers. Since our model adopts GQA, where the number of KV heads is smaller than the number of Query heads, we handle the dimension mismatch by applying a channel-wise repeat technique. Specifically, the key-value projection matrices for the GDN branch are initialized by repeating the channels of the original pre-trained key-value weights to align with the required GDN internal dimensions. This ensures that the LA states initially capture similar key-value interactions as the original attention heads.

Layer-wise Hidden State Alignment. With the parameters initialized, we freeze the pre-trained backbone and optimize only the newly introduced hybrid attention layers. The objective is to align the output hidden states of each hybrid attention layer with those of the original pre-trained FA layer. We employ the Mean Squared Error (MSE) loss between the hidden state representations of the original FA layer ($\mathbf{H}_{FA}^{(l)}$) and the hybrid layer ($\mathbf{H}_{Hybrid}^{(l)}$) for each layer l:

$$\mathcal{L}_{align} = \sum_{l=1}^{L} \|\mathbf{H}_{FA}^{(l)}(x) - \mathbf{H}_{Hybrid}^{(l)}(x)\|_2^2. \tag{13}$$

This layer-wise MSE alignment ensures that the hybrid architecture locally mimics the feature space of the pre-trained model, preventing error accumulation in deeper layers and providing a robust foundation for subsequent global distillation.

B.2.2 Stage 2: Global Logits Distillation

In the second stage, we unfreeze the entire model and perform global knowledge distillation to align the final output distribution of the hybrid student model with the original pre-trained teacher model. Unlike Stage 1, which focuses on intermediate layer consistency, this stage targets the final logits to ensure global semantic coherence. We combine the KL divergence loss on the final logits with a standard cross-entropy loss on the ground truth labels:

$$\mathcal{L}_{KD} = D_{KL}\left(P_{teacher}(\cdot|x) \parallel P_{student}(\cdot|x)\right), \tag{14}$$

where $P_{teacher}$ and $P_{student}$ are the softmax probabilities of the teacher and student models, respectively. This stage allows the hybrid model to recover any minor performance degradation caused by the linear approximation and gate modulation, ensuring that the global linguistic capabilities of the pre-trained backbone are fully preserved.

B.2.3 Stage 3: Long-Context Fine-tuning

By this point, the hybrid architecture has been well-initialized and its output distribution aligned with that of the teacher model through the preceding stages. We therefore adopt standard supervised fine-tuning with the Next Token Prediction (NTP) objective in the final stage, using a longer context length than in the first two stages to consolidate long-context capabilities.

$$\mathcal{L}_{NTP} = -\sum_{t} \log P_{student}(x_{t+1}|x_{1:t}). \tag{15}$$

C More Details of Experiments Setup

C.1 Training Details

In our ablation studies, we follow the default settings of existing work (Chen et al., 2026). Some training configurations within each stage are summarized in Table 6.

C.2 Evaluation Details

The following are the evaluation benchmarks used in the main experiments.

Table 6: Default training details of three training stages, which follow the HypeNet (Chen et al., 2026).

Stage	Batch Size	Context Length	Training Steps (Tokens)	LR scheduler	Learning Rate
1	32	512	20K (0.3B)	Cosine	1e-3 $\rightarrow$ 1e-5
2	96	512	20K (1.0B)	Cosine	1e-4 $\rightarrow$ 1e-5
3	128	16384	500 (1.0B)	Constant	1e-5

Long-Context Retrieval (RULER). We employ the RULER benchmark (Hsieh et al., 2024) to evaluate the model's ability to retrieve information from extended contexts. Specifically, we focus on two categories within RULER: *Single-Key* (1/2/3) and *Multi-Key* (1/2) retrieval tasks, which test the model's precision in locating single or multiple distinct pieces of information embedded in long sequences. We evaluate performance at five context lengths: 16K, 32K, 64K, 128K, and 256K tokens. Given that Qwen3-1.7B has a native context window of 32K, we categorize the evaluation results into two metrics:

- **Native Context Performance:** The average accuracy across 16K and 32K lengths, reflecting the model's baseline capability within its pre-trained window.
- **Extended Context Performance:** The average accuracy across 64K, 128K, and 256K lengths, measuring the model's extrapolation ability and the effectiveness of our hybrid architecture in handling sequences beyond the native limit.

General Reasoning and Knowledge. To ensure that hybrid attention does not compromise the model's general reasoning capabilities in practical scenarios, we evaluate on a broad suite of benchmarks spanning two tiers: standard academic benchmarks commonly used in prior work, and more challenging evaluations widely adopted for assessing frontier large language models.

- **Basic Competency (Easy):** Following prior works (Dao & Gu, 2024; Zhao et al., 2026; Chen et al., 2026), we include six widely used datasets that assess basic language understanding and common sense reasoning: *ARC-Challenge* (norm score) (Clark et al., 2018), *ARC-Easy* (Clark et al., 2018), *HellaSwag* (norm score) (Zellers et al., 2019), *Winogrande* (Sakaguchi et al., 2019), *PIQA* (Bisk et al., 2019), and *LAMBADA-OpenAI* (Paperno et al., 2016). These tasks serve as a sanity check for the model's foundational capabilities.
- **Advanced Reasoning (Hard):** To probe deeper cognitive abilities, we evaluate on four challenging benchmarks requiring complex logical deduction, mathematical reasoning, and code generation: *MMLU* (Hendrycks et al., 2021), *GSM8K* (Cobbe et al., 2021), *MBPP* (Austin et al., 2021), and *BBH* (Suzgun et al., 2022). This tier critically tests whether the efficiency gains from our hybrid design come at the cost of high-level reasoning performance.

All evaluations are conducted using the lm-evaluation-harness framework (Gao et al., 2024), adhering to its default configurations for each benchmark. This includes standard zero-shot or few-shot settings and greedy decoding as specified by the original protocols, unless otherwise noted.

D More Details of Various Hybrid Architectures

To comprehensively evaluate the efficacy of different hybridization strategies, we construct and compare three representative architectures: layer-wise hybrids, token-wise hybrids, and head-wise hybrids (Figure 5). All models incorporate the structural optimizations proposed in our main paper, including the addition of a gating branch to Full Attention (FA) layers with scaling mechanisms replacing RoPE, and the integration of RoPE positional encoding into Gated DeltaNet (GDN) layers.

Layer-wise Hybrid Architectures. We explore two distinct granularities for layer-wise mixing, guided by the layer selection masks derived from HypeNet. (a) **FA & LA:** Each transformer layer is configured as either an FA block (FA + MLP) or a Linear block (GDN + MLP). The HypeNet-derived mask determines which layers retain FA capabilities. (b) **LA-FA & LA:** Each layer is configured as either a Composite block (GDN + FA + MLP) or a Linear block (GDN + MLP). In this setting, the

HypeNet mask identifies layers that require the combined power of both attention mechanisms, while others rely solely on GDN for efficiency.

Token-wise Hybrid Architectures. As illustrated in Figure 5, our token-wise design conceptually assigns FA (implemented as Sliding Window Attention, SWA) to tokens within the recent window and LA to those outside it. For implementation simplicity, we realize this by processing all tokens through the GDN branch for global context modeling while additionally attending to recent tokens within the sliding window via FA to capture high-precision local dependencies. We consider two variants of this design. The first applies 4K-context SWA combined with GDN uniformly across all layers. The second adopts a layer-interleaved strategy inspired by Liger (Lan et al., 2025): 75% of the layers use SWA (window size 128) with GDN, while the remaining 25% combine full FA with GDN. Unlike Liger, which employs pure FA in its global layers, our variant retains GDN alongside FA to preserve token-wise hybridization throughout all layers.

Although this design can be categorized as token-wise mixing due to the varying receptive fields across token positions, the way features from the two branches are combined is essentially identical to the head-wise mixing approach. A more canonical form of token-wise mixing operates at the KV cache level during softmax attention computation, where different tokens attend to KV caches of varying lengths or compositions (Meng et al., 2026a; Zhang et al., 2025). However, our preliminary experiments reveal that such methods struggle to acquire long-context capabilities through transfer learning when the pretraining context window is orders of magnitude shorter than the target evaluation length. We therefore defer their comparison to future work and do not include them in this paper.

Head-wise Hybrid Architectures. We define two sub-paradigms for head-wise hybridization: **head-wise mixing**, where each head simultaneously computes both FA and GDN branches and fuses their outputs, and **head-wise selection**, where each head is exclusively assigned to either the FA or GDN branch based on a learned or heuristic criterion. For head-wise mixing, we implement a variant inspired by (Dong et al., 2024) that processes the input through parallel FA and GDN branches, each operating over the full set of heads, and fuses their outputs at the feature level within every layer. For head-wise selection, existing work has explored per-head hybridization between FA and sparse attention (Xiao et al., 2024a; Tang et al., 2026). We therefore use HydraHead as the representative of this paradigm that adopts linear attention instead.

E More Results of Hybrid Architecture Comparison

We perform a detailed comparison of various typical hybrid architectures in Table 1. Table 7 summarizes the long-context evaluation results on the RULER benchmark, covering both Single and Multi-key scenarios across context lengths ranging from 16K to 256K tokens. Our method achieves the best performance in nearly all settings, demonstrating robust and broad superiority in long-context modeling. Tables 8 and 9 present the general capability comparisons. Our approach significantly surpasses layer-wise hybrid models, with particularly notable gains in mathematical reasoning and code generation. While other head-wise and token-wise hybrid variants achieve stronger general performance than our model, they suffer from severely compromised long-context capabilities. Taken together, our hybrid design achieves the most favorable trade-off between general competence and long-context proficiency.

F Head-Selection Details and Additional Interpretability Analysis

This appendix specifies the retrieval instantiation of Section 2.1, extends the introductory analysis of head-level variation, and validates the head ranking before conversion.

F.1 Retrieval Calibration and Estimator Implementation

Calibration inputs. We instantiate the capability-directed procedure in Section 2.1 using five NIAH probes from RULER (Hsieh et al., 2024): three single-key and two multi-key variants. Each probe uses eight clean–counterfactual pairs at 4K context length. The scores are computed on the pretrained

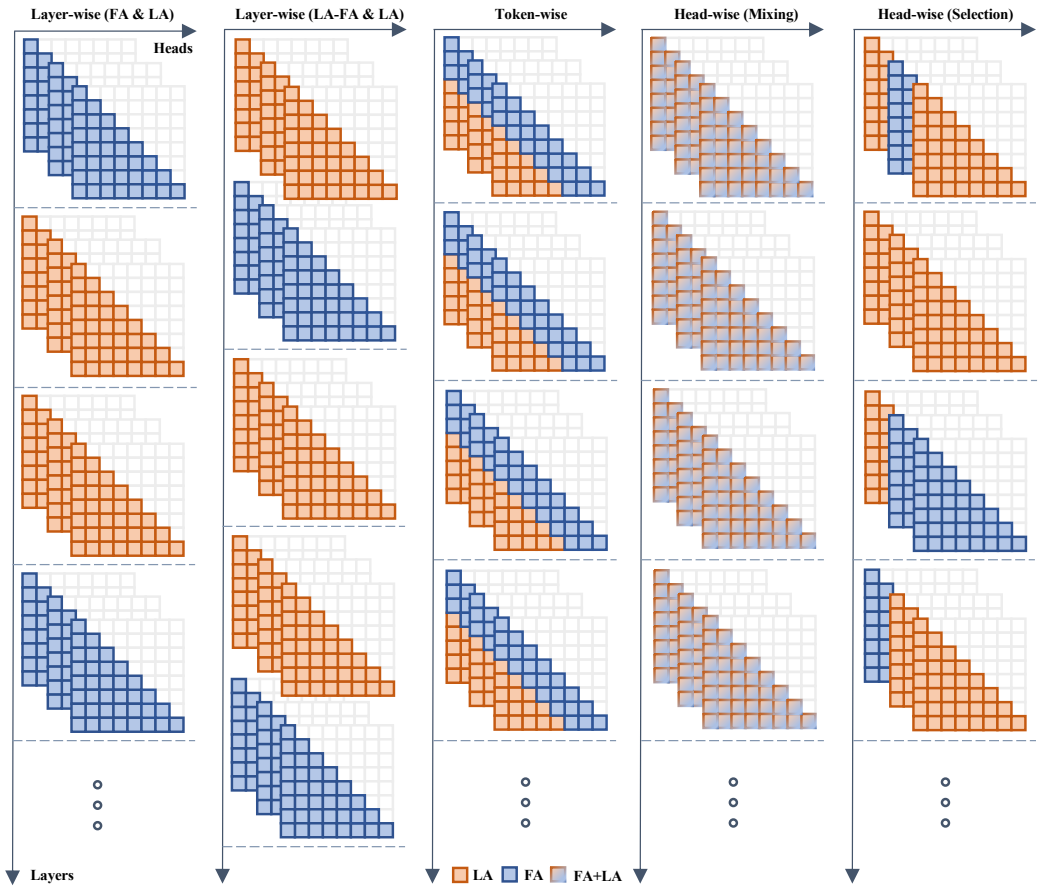


Figure 5: Visualization of some typical hybrid methods. The hybrids can be categorized in terms of layer-wise, token-wise, and head-wise.

Table 7: Comparison of various hybrid architectures in the **RULER** benchmark. Cache denotes the relative key-value cache under a 16K context length, where the standard FA cache is normalized to 1.0. Light. denotes the lightning attention (Qin et al., 2024). * and ‡ indicate the hybrids are similar to HypeNet (Chen et al., 2026) and Liger (Lan et al., 2025), respectively. † indicates that an FA layer is inserted after every 3 LA layers. HW denotes Head-wise.

Model	LA	RULER Single					RULER Multi-Key				
		16K	32K	64K	128K	256K	16K	32K	64K	128K	256K
Layer-wise Hybrid Transformers											
FA & LA*	Light.	92.13	86.00	88.87	84.47	**81.67**	28.90	20.80	**32.70**	22.70	17.70
FA & LA†	Light.	87.67	84.40	80.33	69.40	53.53	36.70	26.70	26.00	14.60	12.80
FA & LA	GDN	62.87	32.40	15.67	4.20	1.20	25.10	19.90	10.30	4.80	1.50
LA-FA & LA	GDN	73.13	43.87	17.27	5.20	1.93	23.00	15.00	12.80	5.70	2.00
Token-wise & Head-wise Hybrid Transformers											
Token-wise	GDN	28.07	13.47	6.07	3.00	2.13	21.60	10.50	3.40	2.40	1.50
Token-wise‡	GDN	97.87	94.20	80.87	56.13	39.33	29.00	23.40	15.70	9.60	7.80
HW Mixing	GDN	98.33	88.07	77.73	61.73	41.80	46.10	28.60	21.30	14.30	8.00
HydraHead	GDN	**98.60**	**98.33**	**94.60**	**89.00**	78.87	**41.40**	**32.80**	32.30	**26.60**	**23.20**

Table 8: Comparison of various hybrid architectures in the general benchmark (**Easy**). Cache denotes the relative key-value cache under a 16K context length, where the standard FA cache is normalized to 1.0. Light. denotes the lightning attention (Qin et al., 2024). * and ‡ indicate the hybrids are similar to HypeNet (Chen et al., 2026) and Liger (Lan et al., 2025), respectively. † indicates that an FA layer is inserted after every 3 LA layers.

Model	LA	ARC-C	ARC-E	HellaSwag	LAMBADA	PIQA	Winogrande
Layer-wise Hybrid Transformers							
FA & LA*	Light.	41.38	71.13	58.81	52.88	72.74	61.40
FA & LA†	Light.	43.52	72.01	59.20	50.86	72.85	63.30
FA & LA	GDN	39.85	70.75	58.05	47.25	71.76	61.40
LA-FA & LA	GDN	41.21	70.24	58.46	49.93	72.03	60.30
Token-wise & Head-wise Hybrid Transformers							
Token-wise	GDN	**46.50**	75.38	**62.58**	**58.49**	74.05	63.38
Token-wise‡	GDN	45.65	75.34	62.43	58.28	**74.27**	**64.25**
Head-wise Mixing	GDN	43.60	**76.09**	61.97	57.33	73.61	64.01
HydraHead	GDN	42.83	76.05	61.82	56.32	73.88	61.80

Qwen3-1.7B checkpoint in `fp16`. These retrieval probes supply the head-selection calibration; the general-reasoning benchmarks assess the converted model.

Counterfactual construction. For each clean input x, we replace the needle value with a randomly generated counterfactual of the same type and token length. The haystack, query, and surrounding structure are unchanged. The paired inputs therefore differ in the answer-bearing content while preserving the structure and length of the retrieval problem.

Multi-token readout. Let $\mathcal{A}$ denote the answer-token positions, and let a_j^+ and a_j^- be the correct and counterfactual answer tokens at position j. Writing z_j for the output logits at that position, we extend the single-token readout in Equation equation 5 to

$$m(x) = \frac{1}{Z} \sum_{j \in \mathcal{A}} \lambda^j \left(z_j[a_j^+] - z_j[a_j^-] \right), \qquad Z = \sum_{j \in \mathcal{A}} \lambda^j, \tag{16}$$

with $\lambda = 0.9$. The weighting emphasizes earlier answer positions. Token identities and position weights are held fixed when evaluating the clean, counterfactual, and patched executions of a pair.

Table 9: Comparison of various hybrid architectures in the general benchmark (**Hard**). Cache denotes the relative key-value cache under a 16K context length, where the standard FA cache is normalized to 1.0. Light. denotes the lightning attention (Qin et al., 2024). * and ‡ indicate the hybrids are similar to HypeNet (Chen et al., 2026) and Liger (Lan et al., 2025), respectively. † indicates that an FA layer is inserted after every 3 LA layers.

Model	LA	MMLU	BBH	MBPP	GSM8k
Layer-wise Hybrid Transformers					
FA & LA*	Light.	48.80	13.79	10.00	6.60
FA & LA†	Light.	39.70	9.77	7.60	6.67
FA & LA	GDN	46.67	13.58	2.40	3.11
LA-FA & LA	GDN	48.32	14.53	4.40	6.44
Token-wise & Head-wise Hybrid Transformers					
Token-wise	GDN	**56.65**	**47.87**	**34.60**	**50.11**
Token-wise‡	GDN	53.17	27.65	29.20	36.63
Head-wise Mixing	GDN	54.46	35.94	29.20	32.68
Ours	GDN	49.86	29.15	22.00	23.12

Receiver and sender execution. Receiver effects use the normalization in Equation equation 1. Every downstream attention output is fixed to its clean value, while downstream MLPs are recomputed. This measures the head's effect through the residual stream and downstream MLP computation. Sender effects restrict the intervention to selected receiver pathways and use the same readout normalization. Newly identified senders become receivers for the next upstream-tracing round. In the retrieval analysis, this tracing takes roughly two rounds.

Candidate screening and computation. The reported experiments use full activation patching, with a layer-level screen followed by head-level refinement within the surviving layers. The layer screen patches all heads in a layer together. No gradients or attribution approximations are used for the reported head effects. Calibration-set size and context-length sensitivity are evaluated in Appendix F.3.

Allocation score and diagnostic thresholds. Equation equation 2 combines receiver and sender effects with probe consistency; the threshold in that definition is $\tau = 0.01$. The head-localization diagnostics in Appendix F.3 additionally report magnitude-based categories using cutoffs of 0.01 and 0.005. The FA budget determines how many ranked heads are retained.

F.2 Functional Diversity of Heads Within a Layer

The analyses in Figure 2 compare head-level readouts with layer-level output similarities. Here we examine within-layer variation in more detail. On Qwen3-1.7B, cosine similarities between head outputs span a wide range within individual layers (Figure 6, left), with near-orthogonal pairs in several layers (Figure 7). The corresponding layer-output similarity matrix has the smoother block-wise structure shown in Figure 2b.

Logit-lens readouts provide a complementary view. In middle and later layers, the 16 heads produce distinct top-1 tokens and divergent vocabulary readouts (Figure 6, right). These measurements show variation within a layer at both the representation and readout levels, motivating the head-wise granularity. Appendix F.3 examines the behavioral effects of intervening on the heads.

F.3 Validation of Interpretability-Guided Head Selection

We characterize the retrieval-head effects on the pretrained Qwen3-1.7B model (28 layers, 16 query heads per layer), using the calibration procedure in Appendix F.1. The analyses examine ranking stability, within-layer concentration, and the behavioral effects of intervening on high-ranked heads. Figure 10 shows the allocation masks used in the converted-model comparison of Table 2.

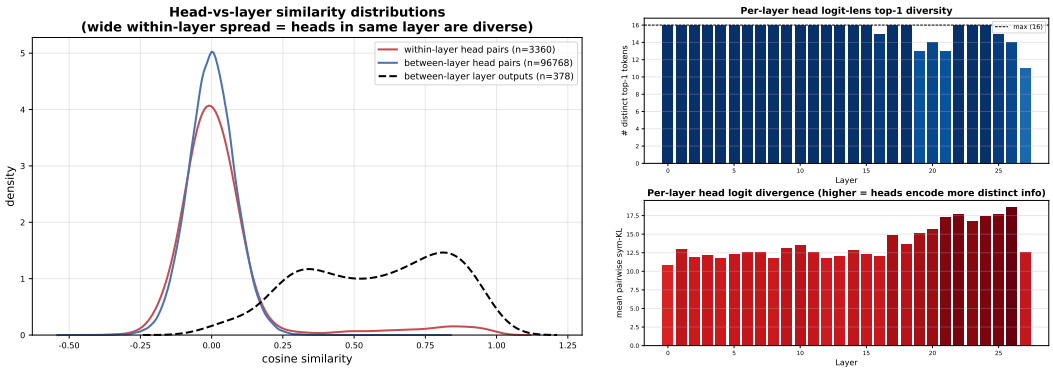


Figure 6: Head-output and readout diversity on Qwen3-1.7B. **Left:** Cosine-similarity distributions for within-layer head pairs, between-layer head pairs, and full layer-output pairs. **Right:** The number of distinct top-1 logit-lens tokens across the 16 heads in each layer and their mean pairwise symmetric KL divergence.

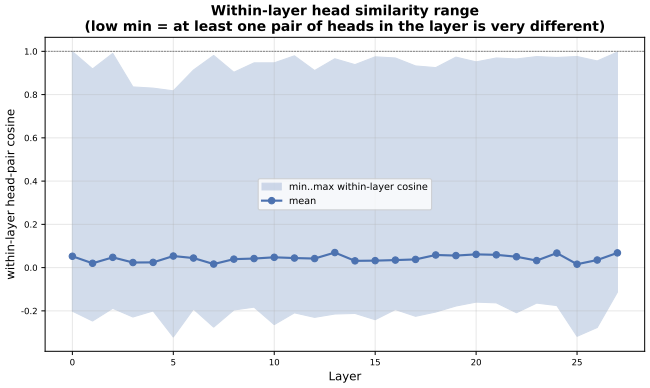


Figure 7: Per-layer minimum, mean, and maximum cosine similarity among head outputs on Qwen3-1.7B. Several layers contain pairs with near-zero cosine similarity. Full layer-output similarities are shown in Figure 2b.

Calibration and context-length stability. Figure 8 compares rankings obtained from different calibration-set sizes against a 32-sample reference. Spearman correlation reaches approximately 0.921 at six samples and saturates around twelve. Across localization context lengths, top-32 set overlap remains stronger than exact rank agreement at some intermediate lengths. The probes also differ: number-needle probes show greater agreement than the UUID-needle probes. These results motivate examining both set overlap and rank correlation rather than treating them as interchangeable stability measures.

Concentration within layers. Head importance is uneven within individual layers: the mean per-layer Gini coefficient is 0.622, with values from 0.399 to 0.915. The context-averaged, magnitude-based categorization identifies 29 heads at or above the 0.01 threshold and 407 below 0.005, out of 448 heads in total. Table 10 lists layers containing both high- and low-importance heads. This within-layer variation gives head-wise allocation more flexibility than assigning one mechanism to every head in a layer.

Intervention validation. We replace head outputs with their counterfactual-run activations during prefill, then generate greedily and measure needle-retrieval accuracy. Heads are selected in decreasing order of estimated importance; controls replace the same number of randomly chosen heads over multiple runs. As shown in Figure 9b, intervening on the top few ranked heads reduces accuracy sharply, while matched-size random selections have substantially smaller effects. The experiment

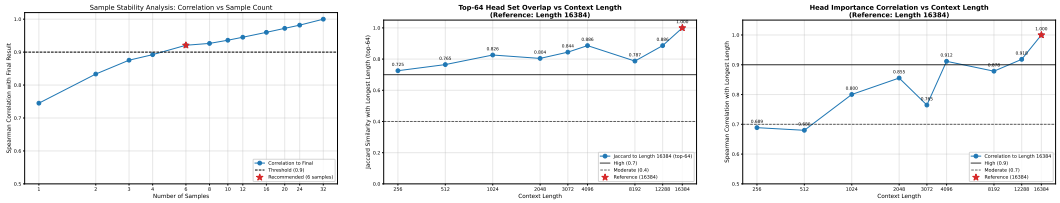


Figure 8: Head-ranking stability on Qwen3-1.7B. **Left:** Spearman correlation with a 32-sample reference reaches approximately 0.921 at six calibration samples and saturates around twelve. **Middle and right:** Top-32 Jaccard overlap and Spearman correlation across localization context lengths, relative to the 16K reference. Set overlap and exact rank agreement measure different aspects of stability.

Table 10: Layers containing both high- and low-importance heads on Qwen3-1.7B, using the magnitude-based categories defined in the text.

Layer	27	23	20	21	25	26	19	22	24
# high-importance heads	6	4	3	3	3	3	2	2	2
# low-importance heads	9	12	11	10	13	12	13	14	13

tests whether the ranking identifies components that influence retrieval generation, complementing the allocation comparison in Section 3.3.

The allocation score guides which heads receive FA under a specified budget. The budget itself is evaluated after conversion: Appendix G.3 compares 3:1, 7:1, and 9:1 GDN-to-FA ratios, including the variant that retains at least one FA head per layer.

G Additional Experimental Results

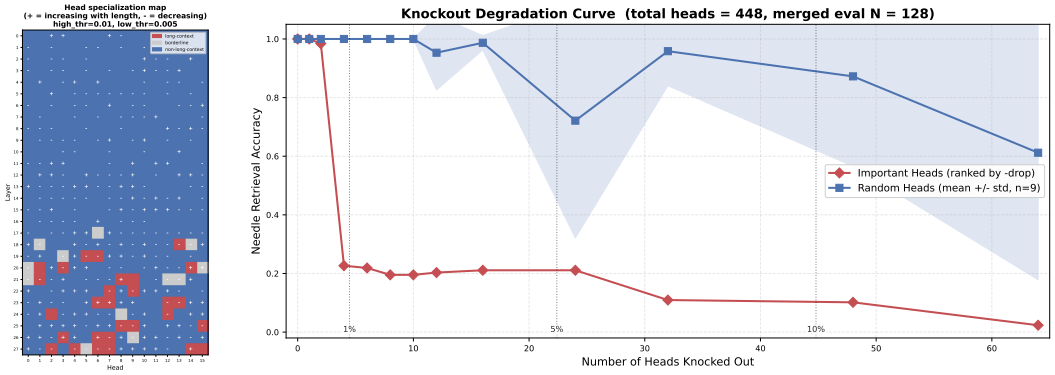
G.1 Structural Components Study

We conduct a systematic ablation study to dissect the contribution of each architectural component in our head-wise hybrid design, focusing on the interaction between the FA and GDN heads.

Experimental Setup: In this series of experiments, we adopt a **fixed head-wise allocation strategy** to isolate the impact of structural modules from the variability of dynamic routing. Specifically, for every transformer layer, we assign the first 75% of attention heads to the GDN branch and the remaining 25% to the FA branch (Figure 10 (a)). This static configuration ensures a consistent computational budget across all variants, allowing for a fair comparison of how different internal mechanisms affect model expressiveness and stability.

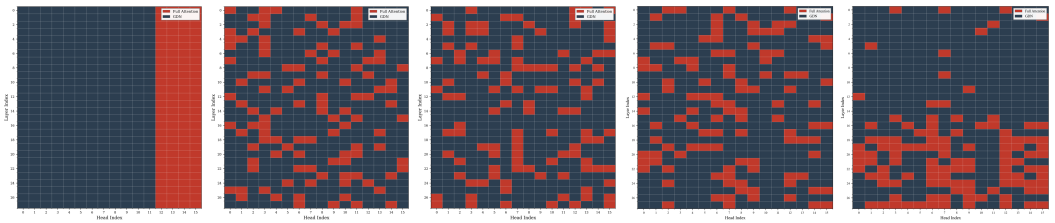
Progressive Module Integration. We start from a minimal hybrid baseline and progressively introduce key modules to observe their individual and combined effects. The variants are defined as follows:

- **Base Hybrid**: A naive combination of FA and GDN heads without any specialized stabilization or feature alignment techniques. Note that the normalization of output (see Equation 3) is used.
- **+ FA NoPE & Scale**: We remove positional embeddings (NoPE) for the FA branch and use position-related scaling instead. The scale hyperparameter is set to 500.
- **+ GDN RoPE**: We apply location encoding to the GDN branch to enhance content-aware capabilities.
- **+ FA Gate**: We incorporate a gating mechanism within the FA heads to dynamically modulate their contribution based on input context.
- **+ GDN MHA**: Unlike FA, which uses GQA to reduce key-value cache, we use MHA in GDN to better utilize the multi-head mechanism to capture semantic associations.



(a) (b)

Figure 9: Head localization and intervention validation on Qwen3-1.7B (448 query heads). **(a)** Each cell represents a head, with categories defined by retrieval importance and $+/-$ markers indicating changes with context length. **(b)** Replacing importance-ranked head outputs with counterfactual activations during prefill reduces retrieval accuracy more sharply than matched-size random selections; random controls show mean $\pm$ standard deviation.



(a) **(b)** **(c)** **(d)** **(e)**

Figure 10: Comparison of distribution results of 5 head screening strategies: (a) Fixed Allocation; (b) Layer-wise Random Selection; (c) Global Random Selection; (d) Layer-wise Interpretability Screening; (e) Global Interpretability Screening.

- **+ Query Decomposition**: Finally, we decompose the query projection matrices for FA and GDN heads. This avoids potential optimization bottlenecks in practice that may arise from numerical precision limitations (Micikevicius et al., 2018) and implicit gradient conflicts between the two distinct heads.

Results and Analysis. Table 11 summarizes the performance variations resulting from the incremental addition of each module. We observe that replacing the positional encoding in the FA branch with scale modulation significantly improves the hybrid model's long-context capabilities. In both Single-key and Multi-key tests, performance within the original context length increased by over 10%, while extended context lengths improved from negligible gains to a substantial level. However, this modification reduces the positional awareness of the FA branch, leading to a decline of more than 3% in general reasoning performance. Adopting a Multi-Head Attention (MHA) configuration for the GDN branch helps recover part of this lost general capability. The full configuration further decomposes the query projection matrices, resulting in the splitting of QKV projection matrices for both branches. This configuration achieves optimal performance on the RULER Single test while maintaining robust results in Multi-Key and general benchmarks. This advantage also holds consistently under the **interpretability-guided head selection**[3] setting (Table 12). Therefore, we denote this version model as **Basic Model**.

[3]It's denoted as Global Interpretability Screening in Section 3.3.

Table 11: Ablation study on structural components of the hybrid architecture under the **fixed head-wise allocation strategy**. Each module in each row is added based on the previous row.

Model	RULER Single		RULER Multi-Key		General Reasoning	
	(Native)	(Extended)	(Native)	(Extended)	(Hard)	(Easy)
Base Hybrid	35.33	2.96	9.90	1.60	**31.14**	62.42
+ FA NoPE & Scale	64.73	40.58	**31.65**	8.47	27.87	62.22
+ GDN RoPE	72.80	56.38	24.00	14.00	26.36	62.09
+ FA Gate	76.07	60.89	31.05	**15.07**	26.98	62.60
+ GDN MHA	79.97	54.27	31.35	14.17	29.66	62.42
+ Query Decomposition	**85.63**	**62.62**	27.35	13.67	28.65	**62.59**

Table 12: Query decomposition yields more pronounced gains under **interpretability-guided head selection** than under fixed allocation.

Model	RULER Single		RULER Multi-Key		General Reasoning	
	(Native)	(Extended)	(Native)	(Extended)	(Hard)	(Easy)
Basic Model	98.70	81.73	38.35	25.93	31.70	62.68
w/o Query Decomposition	91.30	79.43	46.85	25.77	31.79	62.78

G.2 Feature Fusion Designs

We systematically investigate feature fusion strategies based on a head-wise hybrid architecture. The core challenge lies in reconciling the different statistical properties of features generated using softmax and linear attention mechanisms—particularly differences in amplitude and variance. Directly concatenating these heterogeneous features often imposes significant optimization pressure on subsequent output projection layers, potentially leading to training instability or performance degradation.

Experimental Setup. We conduct experiments under the **interpretability-guided head selection** strategy, where heads are selected based on their causal importance scores derived from mechanistic interpretability analysis. Unlike the fixed strategy, this global selection allows for heterogeneous layer-wise distributions; specific layers may exhibit extreme configurations, such as being composed entirely of GDN/FA heads, while others retain a higher concentration of GDN/FA heads. This setup creates a more complex and realistic optimization landscape, enabling us to rigorously assess the robustness of different fusion mechanisms in handling significant distributional shifts between attention branches.

Feature Fusion Variants. To systematically evaluate the impact of feature alignment strategies, we construct three variants based on the Basic Model, progressively introducing mechanisms to handle the distributional heterogeneity between FA and GDN branches:

- **Direct Concatenation (w/o Norm):** This baseline variant removes all normalization layers, directly concatenating the raw output features from FA and GDN heads.
- **Head-wise Scale Modulation:** Building upon a standard normalization step (e.g., RMSNorm) applied to each branch independently, this variant introduces learnable scalar coefficients (see Equation 4). This allows the model to statically adjust the relative contribution magnitude of each head type during training.
- **Head-wise Gated Competition:** This variant replaces static scaling with a dynamic gating mechanism. A lightweight gate branch takes the original input sequence representation $\mathbf{x}$ as input and produces head-wise modulation weights via a softmax activation, scaled by the total number of heads H to preserve the overall feature magnitude. The final fusion is performed just like head-wise scale modulation. By enforcing a sum-to-one constraint across the FA and GDN components for each head position, this design introduces explicit competition, enabling context-aware selection between FA and GDN.

Table 13: Comparison of feature fusion variants under the **interpretability-guided global selection**.

Model	RULER Single		RULER Multi-Key		General Reasoning	
	(Native)	(Extended)	(Native)	(Extended)	(Hard)	(Easy)
Basic Model	**98.70**	81.73	38.35	25.93	**31.70**	**62.68**
w/o Norm	85.07	71.36	**41.90**	23.93	30.62	62.58
Head-wise Scale	98.47	**87.49**	37.10	**27.37**	31.03	62.12
Head-wise Gate	95.80	67.16	41.85	26.87	28.24	61.90

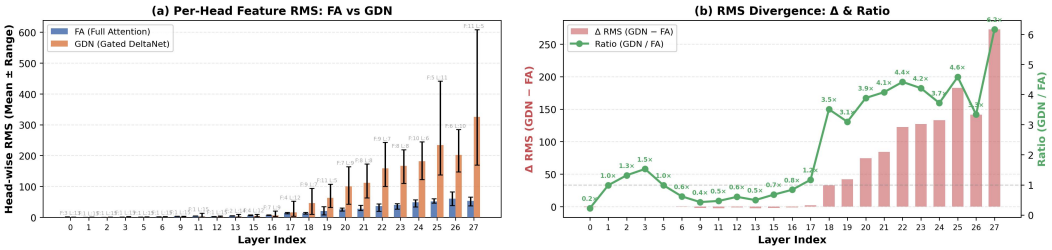


Figure 11: Per-head feature RMS statistics of FA and GDN branches before normalization (Default Model). (a) Layer-wise RMS comparison with min–max range. (b) Divergence measured by Δ RMS and ratio.

Results and Analysis. We present the comprehensive experimental results in Table 13. Our analysis focuses on two key aspects: the necessity of feature normalization, and the trade-offs between static scaling and dynamic gating.

Impact of Feature Normalization. The results reveal a consistent and substantial degradation across most evaluation dimensions. On RULER Single-key retrieval, removing normalization leads to a performance drop exceeding 10% at both the original and extended context lengths, indicating that the feature distributions from heterogeneous attention heads are sufficiently misaligned to severely impair retrieval fidelity when naively combined. On challenging general reasoning benchmarks, a moderate but consistent degradation of approximately 1% is observed. The sole exception is Multi-Key retrieval at the original context length, where direct concatenation yields a 3.55% improvement—a counterintuitive result that warrants further investigation. One plausible explanation is that in the Multi-Key setting, the model must attend to multiple dispersed positions simultaneously, and the unnormalized feature mixing may inadvertently amplify certain cross-head interactions that facilitate multi-location tracking, at the cost of degraded single-location precision. Nevertheless, given the overwhelming degradation across all other metrics, this isolated gain does not justify removing normalization. We conclude that feature normalization is a critical design component for ensuring robust and universal performance across diverse head allocation strategies and evaluation scenarios.

Scale Modulation vs. Gated Competition. We compare the two advanced fusion variants, Head-wise Scale Modulation and Head-wise Gated Competition, across the full evaluation suite. The results reveal a clear and consistent advantage for the Scale Modulation variant: it outperforms Gated Competition on nearly all metrics, with the sole exception of Multi-Key retrieval at the original context length. The most striking gain is observed in long-context extension scenarios, where Scale Modulation achieves a 20% improvement over Gated Competition on Single-key retrieval tasks, demonstrating that static, learnable coefficients provide substantially more stable representations for modeling long-range dependencies under length extrapolation. Furthermore, compared to the Basic model, Scale Modulation delivers consistent improvements across all extended-context evaluations, confirming that the head-wise scaling mechanism generalizes robustly beyond the training distribution. The only setting where Gated Competition holds an edge—Multi-Key at the original context length—suggests that dynamic, context-aware gating may offer localized benefits when multiple retrieval targets are densely distributed within the training-length regime, but this advantage is not preserved under length expansion. Given the overwhelming and consistent superiority of Scale Modulation across both general reasoning and long-context benchmarks, we adopt Head-wise Scale Modulation as the **Default Model** for all subsequent experiments.

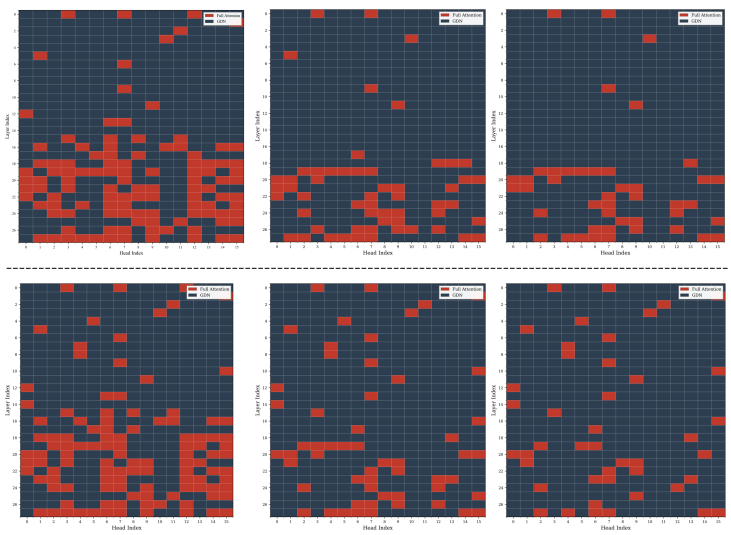


Figure 12: Comparison of head selection distributions under varying GDN-to-FA ratios. The top row illustrates the distribution obtained by direct screening based on head interpretability scores. The bottom row shows the distribution with an additional constraint that ensures at least one FA head is retained in each layer, while still prioritizing high-scoring heads.

Feature Distribution Differences. Figure 11 reports the per-head feature RMS statistics of the FA and GDN branches prior to head-wise fusion. In shallow layers (0–17), the two branches exhibit comparable magnitudes (GDN/FA ratio within [0.2,1.5]), partly because shallow layers are dominated by GDN heads (Figure 10(e)), leaving few FA heads as a contrasting reference. In deeper layers (18–27), however, a pronounced scale mismatch emerges, with GDN RMS reaching up to 6.2× that of FA at layer 27. This divergence stems from the norm-awareness asymmetry between the two paradigms: softmax in FA constrains output magnitudes via exponential normalization, whereas linear attention in GDN lacks such regularization. The observed mismatch motivates independent RMSNorm for each branch before fusion, preventing either branch from dominating the combined representation due to scale disparities alone.

G.3 Impact of Higher Hybrid Ratios

We investigate the performance trends under higher GDN-to-FA mixing ratios to further explore the efficiency potential of our hybrid architecture. While a 3:1 ratio provides a balanced trade-off, increasing the proportion of GDN can significantly reduce memory footprint and computational overhead.

Experimental Setup. We evaluate three mixing ratios, extending from the baseline 3:1 to more aggressive configurations of 7:1 and 9:1. For each ratio, we compare two head selection strategies:

- **Global Interpretability Screening:** Heads are selected strictly based on their global causal importance scores, regardless of their layer distribution. This serves as the baseline to test the raw efficacy of interpretability-guided selection.
- **Constrained Global Screening:** Recognizing that strict global ranking at high sparsity levels may concentrate FA heads in only a few layers—leaving most layers with pure GDN and potentially disrupting hierarchical information flow—we introduce a structural constraint. This strategy ensures that at least one FA head is retained in every layer, while the remaining FA budget is allocated to the highest-scoring heads globally.

This comparison allows us to disentangle the benefits of functional importance from the necessity of structural coverage across network depth. Figure 12 visualizes the resulting head distributions under

Table 14: Performance comparison under varying GDN-to-FA ratios. We evaluate two selection strategies: *Global-Interp* (direct screening by interpretability scores) and *Global-Interp-C* (with a constraint of at least one FA head per layer). All experiments are conducted based on **Basic Model**.

Model	Ratio	RULER Single		RULER Multi-Key		General Reasoning	
		(Native)	(Extended)	(Native)	(Extended)	(Hard)	(Easy)
Global-Interp	3:1	**98.70**	81.73	**38.35**	**25.93**	31.70	62.68
Global-Interp-C	3:1	96.27	**83.91**	36.45	25.20	**31.83**	**62.73**
Global-Interp	7:1	81.40	58.07	17.90	18.67	27.75	62.63
Global-Interp-C	7:1	**88.70**	**81.04**	**26.70**	**22.93**	**29.46**	**62.65**
Global-Interp	9:1	89.12	60.58	18.65	14.50	26.04	62.65
Global-Interp-C	9:1	**90.10**	**72.69**	**21.05**	**18.63**	**28.92**	**62.68**

Table 15: Comparison of layer-wise hybrids with a 3:1 LA:FA ratio (marked with * in Table 1) against our **Basic Model** with a 7:1 ratio. Our model performs *Global-Interp-C* head selection strategy.

Model	Ratio	RULER Single & Multi-Key					Avg	General Reasoning	
		16K	32K	64K	128K	256K		(Hard)	(Easy)
Layer-wise hybrid	3:1	**60.52**	53.40	**60.79**	**53.59**	**49.69**	**55.59**	19.80	59.72
Global-Interp-C	7:1	59.55	**55.85**	57.35	51.04	47.59	54.27	**29.46**	**62.65**

both strategies. As the GDN-to-FA ratio increases, the proportion of layers composed entirely of GDN heads rises steadily, reaching nearly 50% in the extreme 9:1 configuration.

Results and Analysis. Table 14 summarizes the performance results across different hybrid ratios. We can summarize the findings as follows:

Performance Degradation and Structural Resilience. As the GDN-to-FA ratio increases, we observe a consistent decline in general reasoning capabilities, with complex reasoning tasks being particularly sensitive to the reduction of FA heads. In contrast, long-context performance exhibits a fluctuating downward trend. This divergence suggests that while LA can effectively handle local dependencies and simple retrieval, the loss of global receptive fields in FA heads disproportionately impacts multi-step logical deduction. The Global-Interp-C strategy, which ensures at least one FA head per layer, demonstrates significant advantages at higher sparsity levels (7:1 and 9:1). By preventing the complete collapse of FA capacity in certain layers, this constraint maintains the hierarchical information flow essential for deep networks. At lower ratios (3:1), while its impact on long-context tasks varies, it consistently yields improvements in general reasoning, indicating that structural diversity across layers is beneficial even when the overall FA budget is sufficient.

Sparsity Limits and Practical Effectiveness. A direct comparison between the constrained 9:1 configuration and the standard 3:1 baseline reveals the current limitations of our hybrid architecture under extreme sparsity. While the model retains over 80% of its performance on simple long-context tasks, it suffers from significant degradation in more complex scenarios. Specifically, accuracy drops to approximately 60% in multi-key retrieval settings, and general reasoning capabilities on challenging benchmarks decline by about 3%. These results indicate that while the layer-wise constraint effectively prevents structural collapse, the severe reduction in FA heads still compromises the model's ability to handle intricate dependencies and complex logical patterns, suggesting that further optimization is required to bridge the performance gap in extreme sparse regimes. Despite these limitations at extreme sparsity, our model retains a substantial advantage over layer-wise hybrids in practical regimes. At a 7:1 ratio, we achieve long-context performance on RULER (16K–256K) broadly comparable to the layer-wise hybrid at 3:1, with per-length differences within ±3% (Table 15). Meanwhile, it demonstrates a substantial advantage in general reasoning (+9.66% on Hard, +2.93% on Easy), underscoring the effectiveness of head-wise hybridization in preserving strong general-domain capabilities while enabling aggressive LA:FA ratios for long-context extension.

Table 16: Optimized training configuration of three training stages.

Stage	Batch Size	Context Length	Training Steps (Tokens)	LR scheduler	Learning Rate
1	128	2048	3052 (0.8B)	Cosine	1e-3 $\rightarrow$ 1e-5
2	256	2048	7630 (4.0B)	Cosine	1e-4 $\rightarrow$ 1e-5
3	128	16384	477 (1.0B)	Constant	1e-5

Table 17: Comparison of training configuration optimization. † denotes that the results are reproduced by ourselves. New-Config indicates the optimized configuration, which is summarized in Table 16.

Model	RULER Single		RULER Multi-Key		General Reasoning	
	(Native)	(Extended)	(Native)	(Extended)	(Hard)	(Easy)
HypeNet†	**89.07**	**85.00**	24.85	24.37	19.80	59.72
+ New-Config	86.57	76.62	**33.85**	**25.30**	**23.71**	**60.87**
Default Model	98.47	87.49	37.10	27.37	31.03	62.12
+ New-Config	**99.40**	**89.84**	**51.00**	**35.13**	**34.91**	**63.14**

G.4 Training configuration optimization

To maximize the potential of our head-wise hybrid architecture, we conducted a systematic optimization of the training configurations across the three-stage transfer learning pipeline.

Experimental Setup. While maintaining FineWeb-Edu as the core pre-training corpus, we focused on refining the data scale and hyperparameter schedules to enhance both convergence stability and final model performance. We conduct experiments based on the Default Model as denoted before.

Results and Analysis. We optimize the training of the hybrid model transfer learning from multiple dimensions, including data scale and training configuration. The optimized configuration of three training stages is summarized in Table 16. Compared to the original setup, we observe that scaling up the training data for Stage 1 and Stage 2 significantly enhances the efficacy of knowledge transfer. Furthermore, we increase both the training batch size and the sequence length to better accommodate long-context dependencies and stabilize the optimization dynamics across the heterogeneous attention branches. Table 17 presents a detailed comparison of model performance before and after the training configuration optimization. The results highlight two key observations:

Significant Performance Gains for Our Hybrid Model. Under the optimized configuration, our proposed head-wise hybrid architecture demonstrates consistent improvements across all six evaluation benchmarks. Notably, the performance gains are most pronounced in challenging scenarios, such as RULER Multi-key retrieval and complex general reasoning tasks. For instance, accuracy on RULER Multi-key increases by 13.90% at the native context length, indicating a substantial improvement in handling intricate long-context dependencies. This robust improvement underscores the efficacy of our head-wise selection strategy in capturing fine-grained semantic interactions, validating that scaling up data in the alignment and distillation stages effectively empowers the model to internalize heterogeneous attention patterns.

Divergent scaling behavior of HypeNet. In contrast, HypeNet exhibits marginal improvements under the optimized configuration. While it achieves moderate gains exceeding 4.00% on Multi-Key retrieval within the native context length and on challenging general reasoning benchmarks, these improvements are overshadowed by a pronounced degradation on Single-key retrieval tasks, where performance drops by 2.50% to 8.38%. This asymmetry suggests that head-wise hybridization has a higher capacity for absorbing additional training data: the finer-grained FA/GDN assignment allows the model to better exploit the representational complementarity between branches, whereas HypeNet's layer-wise mixing offers less room for the model to redistribute learned representations across attention types.

H LIMITATIONS AND FUTURE WORK

While HydraHead demonstrates promising results in balancing general reasoning and long-context retrieval, several limitations point to directions for future exploration.

Model scale and architecture diversity. Our experiments are primarily conducted on Qwen3-1.7B, a relatively compact dense Transformer. Despite this modest scale, our results provide compelling evidence for the potential of head-wise hybridization as a general design principle—the consistent gains observed across both retrieval-centric and reasoning benchmarks suggest that functional specialization of attention heads is a robust phenomenon worth exploiting at larger scales. Validating this principle on larger-scale models (e.g., 7B, 13B, and beyond) and across diverse architectures—including Mixture-of-Experts and multimodal models—remains an important next step. The functional specialization patterns of attention heads may shift with model scale and modality, and understanding how these shifts interact with our interpretability-guided head selection strategy would further strengthen the generality of the approach.

Training data scale and post-training. The current scaling experiments are limited to approximately 15B tokens, which is modest by contemporary standards. Scaling up the training corpus by an order of magnitude or more, combined with post-training stages such as instruction tuning and reinforcement learning from human feedback, could unlock substantially stronger hybrid models. In particular, it remains to be seen whether head-wise hybridization confers additional benefits in alignment-centric post-training regimes, where the balance between precise instruction following and long-context reasoning becomes especially critical.

Scalability and generality of interpretability-guided selection. The current head selection procedure relies on full activation patching, which requires iterating over all attention heads with separate forward passes per head—tractable at the 1.7B scale but increasingly costly at frontier scales. Nevertheless, even full activation patching remains far cheaper than the alternative of training multiple hybrid models under different head configurations to search for a good assignment. While attribution patching offers a single-backward-pass approximation, its accuracy for this specific use case has not been systematically validated in our experiments. Beyond computational cost, the procedure also depends on carefully designed calibration inputs: the paired counterfactual construction, the choice of readout metric, and the target capability specification (NIAH sub-probes as a proxy for general long-context retrieval) all require domain-informed design decisions that may not transfer directly to new capabilities or task families. That said, this design effort is a one-time cost amortized across all downstream hybrid models built from a given checkpoint, and the resulting selection is stable and transfers across context lengths. Developing more automated or task-agnostic selection protocols would further broaden the applicability of interpretability-guided hybridization.

Gap between interpretability-based importance and deployable FA budget. Our interpretability analysis suggests that only a small fraction of heads are highly important: roughly 6.5% of heads are identified as carrying the most critical retrieval-related signal. However, the ratio experiments indicate that when the retained FA budget is reduced to around 10%, model capability already deteriorates noticeably. This mismatch suggests that the current interpretability signal, while clearly informative, is not yet sufficient to fully determine the minimal FA allocation required for strong end-task performance. One possibility is that the head-importance estimation itself remains incomplete or noisy, and could be improved with better causal analysis or more robust calibration procedures. Another possibility is that the architecture and training pipeline are not yet able to preserve capability under such aggressive compression, even when the most important heads are retained. Addressing this gap may therefore require progress on both fronts: more accurate interpretability-guided selection, and stronger hybrid architectures or optimization strategies that can better exploit extremely sparse FA budgets.

Broader attention mechanism combinations. It is important to emphasize that the head-wise hybridization architecture we propose is not inherently tied to any specific attention mechanism—it is a general architectural paradigm that can accommodate arbitrary combinations of attention variants. In this work, we instantiate the architecture with two representative mechanisms, FA and GDN, as a concrete proof of concept. The architecture naturally extends to a much richer design space. On the efficient attention side, beyond LA variants (e.g., Mamba, RWKV, Linear Transformer), FA can also be combined with sparse attention mechanisms (e.g., sliding window, local attention) or latent-space attention formulations, each offering distinct efficiency–expressiveness trade-offs. More

ambitiously, the architecture supports functional hybridization of more than two attention types—for instance, jointly mixing FA, LA, and sparse attention within a single model, where different heads adopt different mechanisms according to their functional roles. Systematically characterizing the head-level compatibility patterns across this rich design space could lead to more effective and deployment-specific hybrid recipes.

I Related Work

I.1 Mechanistic Interpretability and Head-Level Analysis

Mechanistic interpretability aims to reverse-engineer the internal computations of neural networks through causal and structural analysis. Foundational work establishes a mathematical framework for reasoning about transformer circuits (Elhage et al., 2021) and standardizes activation patching as the primary tool for measuring component-level causal effects (Heimersheim & Nanda, 2024). These techniques have enabled the identification of concrete circuits underlying specific behaviors, such as indirect object identification (Wang et al., 2022) and factual knowledge storage (Meng et al., 2023).

A recurring finding across this line of work is that attention heads serve as meaningful functional units. Voita et al. (2019) identify three specialized head types—positional, syntactic, and rare-word heads—and show that the majority of remaining heads can be pruned with negligible quality loss. Michel et al. (2019) confirm this finding broadly, demonstrating that in many tasks, only a small fraction of heads accounts for most of the model's capability. Clark et al. (2019) further show that individual heads learn to attend to specific syntactic relations, revealing consistent functional specialization. At a deeper level, Wang et al. (2022) decompose the IOI circuit in GPT-2 small into 26 heads organized into seven functional classes—the canonical worked example of head-level circuit discovery—establishing that head-level granularity is sufficient to capture complex multi-step computations.

Collectively, these findings establish attention heads as meaningful and well-studied functional units, and the associated causal tools—particularly activation patching and path patching—provide a principled methodology for quantifying their individual contributions. This body of work directly motivates and supports our investigation: we leverage these tools to analyze the functional structure of a pretrained model and use the resulting insights to guide the design of our hybrid attention architecture.

I.2 Linear Attention

Linear attention addresses the quadratic complexity of softmax attention by replacing the explicit token-to-token similarity computation with a recurrent state update, reducing complexity from $\mathcal{O}(T^2)$ to $\mathcal{O}(T)$. One prominent line of work frames this through the lens of state space models, progressively introducing input-dependent selectivity to overcome the rigidity of fixed dynamics (Gu & Dao, 2023; Dao & Gu, 2024), and more recently extending to complex-valued state spaces with higher-order discretization for richer memory representation (Lahoti et al., 2026). A complementary perspective views linear attention through the lens of fast-weight memory (Katharopoulos et al., 2020), where the key-value state is treated as an associative memory updated via the delta rule for precise credit assignment (Schlag et al., 2021). Subsequent advances integrate gating mechanisms to control memory retention (Yang et al., 2025b), enhance the expressiveness of the gating function itself (Team et al., 2025b), and most recently decouple the update and forget operations for finer-grained memory control (Hatamizadeh et al., 2026). Despite these advances, pure linear attention remains constrained by its fixed-dimensional recurrent state, which limits its capacity for long-range retrieval and exact token-level replication—motivating the hybrid designs that combine linear attention with full attention in a complementary fashion.

I.3 Hybrid Transformers

The pursuit of balancing computational efficiency and representational expressivity has catalyzed a diverse range of hybrid Transformer architectures. Existing efforts can be categorized into three primary paradigms based on the granularity of their hybridization.

Layer-wise Hybridization. This paradigm alternates between different attention kernels across Transformer layers. Many works primarily focus on interweaving full-attention layers with efficient alternatives, such as windowed attention or linear approximations, to mitigate the quadratic memory footprint of long contexts (Fu et al., 2025; NVIDIA et al., 2026; Merrill et al., 2026). For instance, models such as MiMo (Team et al., 2026a), MiniMax-M1 (MiniMax et al., 2025), and Qwen (Team, 2026) systematically balance standard attention with efficient attention at configurable layer-wise ratios. Furthermore, automated strategies like those explored in GLM-5 (GLM-5-Team et al., 2026) utilize Neural Architecture Search (NAS) to discover optimal layer-wise configurations, demonstrating that static interleaving is often suboptimal. Despite the prevalence of layer-wise hybridization in state-of-the-art architectures, prior work has noted that training hybrid models combining linear attention with full attention remains challenging, with many efforts ultimately focusing on full or sparse attention variants (GLM-5-Team et al., 2026; MiniMax et al., 2026; DeepSeek-AI et al., 2026), suggesting that the design space of attention hybridization remains underexplored.

Token-wise Hybridization. To achieve finer granularity, recent studies have proposed blending that operates at the token or sequence level (Meng et al., 2026a; Du et al., 2026; Zhao et al., 2026; Lan et al., 2025). Among them, STILL (Meng et al., 2026a) employs saliency-based gating to decide which tokens warrant the precision of FA versus the efficiency of linear or sparse approximations. While these approaches refine the allocation granularity from the layer level to the token level, they face a potential challenge: maintaining stable and consistent sequence representations across long contexts is non-trivial when different tokens are processed through heterogeneous attention pathways. Few existing works have reported evaluation results beyond 32K context length for such methods. Moreover, the very pursuit of such stability and consistency may inadvertently drive different attention patterns toward convergence, thereby diminishing their ability to complement each other. Additionally, in the case of sparse attention, the set of tokens retrieved varies across queries, necessitating the retention of full KV caches and thus undermining memory efficiency during inference.

Head-wise Hybridization. Recognizing the functional heterogeneity of multi-head attention, recent work has explored head-wise hybridization along two distinct paradigms. The first paradigm performs per-head selection: methods such as DuoAttention (Xiao et al., 2024a) and Elastic Attention (Tang et al., 2026) treat each head as the unit of allocation, using heuristic partitioning or learned routers to designate "critical" heads for FA while assigning the remainder to sparse attention variants. However, since sparse attention and standard FA share the same quadratic dot-product computation core, their complementary benefits are inherently limited—both remain constrained by similar attention patterns, yielding marginal gains rather than a fundamental efficiency breakthrough. The second paradigm performs per-head mixing. For example, Hymba (Dong et al., 2024) processes the input through two parallel branches—a full multi-head FA branch and a Mamba SSM branch—and fuses their output features via element-wise averaging. Although these fusion-based approaches enrich representational capacity, they apply a uniform mixing strategy across all heads without accounting for their inherent functional specialization. In contrast, our architecture leverages interpretability analysis to uncover the inherent functional specialization of different attention heads, and strategically assigns FA and LA to heads based on their complementary strengths. By promoting representational complementarity, it harmonizes the high-precision recall of FA with the constant-cost global context modeling of LA, achieving a synergistic fusion that fully exploits the distinct capabilities of each attention mechanism.

I.4 Taming Transformer to Hybrid Architecture

The prohibitive cost of pre-training hybrid architectures from scratch catalyzes a surge in cross-architecture distillation methods, aiming to transfer the reasoning capabilities of pre-trained Transformers into more efficient counterparts. Early attempts focus on distilling Transformers into pure RNN or State-Space Models (SSMs). For example, T2R (Kasai et al., 2021) and RADLADS (Goldstein et al., 2026) demonstrate the feasibility of architectural conversion. However, MOHAWK (Bick et al., 2025) and LoLCATs (Zhang et al., 2025) later highlight that naive distillation often leads to performance degradation on recall-intensive tasks, as the disparity between quadratic attention and linear-time mixers creates a significant knowledge transfer barrier. To bridge this gap, recent works shift toward multi-stage pipelines and principled initialization strategies. HedgeMamba (Moudgil et al., 2026) proposes a two-stage recipe that first distills softmax attention into a linearized kernel

(Hedgehog) and then uses it to initialize an SSM-based Mamba mixer. Complementary approaches, such as KL-Guided (Li et al., 2025), emphasize the criticality of selective layer conversion to maximize long-context retention. We leverage the HALO (Chen et al., 2026) as our foundation, adapting its efficient distillation procedure to support our novel head-wise hybrid structures. By extending HALO to handle head-level feature distribution shifts through decoupled QKV projections and modulated fusion, our method successfully transitions from standard layer-wise conversion to a more granular, head-specific hybridization strategy.