SHARING MEMORY ADDRESSING ACROSS DEPTH IN HYBRID GATED DELTA NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Does every recurrent memory layer in a language model need its own learned addressing rule? We investigate this question by selectively sharing parameters across the depth of a hybrid Gated DeltaNet language model. Our primary intervention ties query and key projections and their short convolutions across recurrent layers, while preserving independent memory states, value and output transformations, feed-forward networks, and normalization parameters. A value-only sharing control removes exactly the same number of parameters, separating component choice from the amount of parameter reduction. In a 400M-class model trained on FineWeb-Edu, query/key sharing removes 4.40% of unique parameters and incurs a smaller held-out language-modeling penalty than parameter-matched value sharing at a common 10.22B-token checkpoint. Sharing additional update-control parameters has a similarly small penalty, whereas whole-block looping changes the quality–parameter trade-off substantially. However, QK-only tying reduces 4K single-needle retrieval from 80.2% to 37.6%, despite a 0.24% relative held-out perplexity increase; sharing update controls reaches 74.8%. These observations motivate a storage–semantic retrieval separation hypothesis: reusable memory addressing may coexist with depth-specific semantic selection and content transformations. We present this as a mechanistic interpretation to test, not an established functional decomposition. It requires neither identical address semantics across depth nor reduced expanded-layer computation.

1 INTRODUCTION

Hybrid language models combine recurrent, fixed-size memory with attention over explicit token representations. Gated DeltaNet (GDN) updates a matrix-valued memory using input-dependent forgetting and a delta rule (Yang et al., 2025). Kimi Delta Attention (KDA) and Gated DeltaNet-2 develop the state-update mechanism further (Kimi Team, 2025; Hatamizadeh et al., 2026). Much of this progress concerns *how a single memory layer updates its state*. We study a complementary question: *which parameters of that memory mechanism need to be learned independently at every depth?*

A recurrent layer contains several different objects. Queries and keys determine how representations address memory; values specify what is written; the state records evolving contents; and output transformations integrate retrieved information into the residual stream. Conventional architectures independently parameterize these components across depth. That choice does not establish that their depth specificity is equally important. In particular, separate states and separate contents need not require separate addressing *parameters*.

This distinction motivates a selective tying intervention. We share the GDN query/key (QK) pathway across depths, including its short convolutions, without sharing recurrent states or changing the underlying update rule. Each layer receives a different hidden representation, writes different values, and maintains a different memory. A common addressing function therefore need not produce identical keys or queries, or assign the same semantic role to a coordinate at every depth. This is a hypothesis about reusable *computation*, not an assertion that all layers contain the same knowledge.

Parameter sharing is established in recurrent-depth and compressed Transformers (Dehghani et al., 2018; Lan et al., 2019). More directly, LT2 investigates looped architectures with linear attention and Full+GDN hybrids (Deng et al., 2026). Our question is not whether looping or sharing can

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

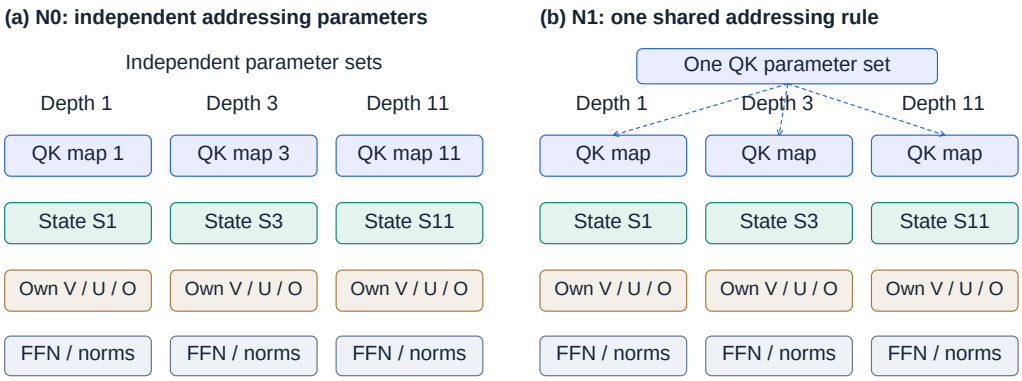


Figure 1: Parameter binding, not state binding. Each column lists components at a selected GDN depth; it is not a computation-flow diagram. N1 replaces independent QK parameter sets with one shared set (dashed connections). QK includes the projections and short convolutions; U denotes update control. States, V/U/O paths, FFNs, and norms remain depth-specific. Intervening Full layers are unchanged and omitted. Equal QK weights do not imply equal QK activations.

work at all. We isolate *what is shared*: an address-side tie is compared with an exactly parameter-matched value-side tie and with stronger hybrid and whole-block ties. All comparisons retain the same expanded depth, widths, and nominal training budget.

We make three contributions. First, we specify a component-level intervention separating shared parameters from independent per-depth memory states. Second, we construct a matched-parameter QK-versus-V comparison and a wider sharing matrix on a controlled GDN+global-attention backbone. Third, we provide a common-checkpoint evaluation protocol covering held-out likelihood, zero-shot tasks, and synthetic retrieval, with explicit parameter accounting and limits on mechanistic inference. The central evidence is a single-seed 400M-scale study; neither larger-scale generality nor faster execution is assumed.

2 Related Work

Gated memory and hybrid attention. GDN combines scalar gated forgetting and delta-rule updates (Yang et al., 2025). KDA and GDN-2 modify memory control and erase/write behavior (Kimi Team, 2025; Hatamizadeh et al., 2026). These architectures motivate separating address, content, and control pathways, but our experiments do not reproduce all of their training recipes or compare their published scores directly. In particular, our global Full-attention component differs from the sliding-window attention of the original GDN-H1 hybrid. The unshared model trained in our own setting is therefore the appropriate baseline.

Sharing and recurrent depth. Universal Transformers reuse a transition across depth (Dehghani et al., 2018); ALBERT studies cross-layer sharing (Lan et al., 2019); and shared feed-forward networks have been investigated in translation (Pires et al., 2023). LT2 extends looping to linear/sparse attention and includes Full+GDN hybrid models (Deng et al., 2026). We do not claim the first use of QK sharing, shared attention, or looped hybrids. We instead use selective tying as a controlled intervention on the addressing and content pathways of a gated delta memory. The value-only control and preservation of independent states distinguish this question from simply maximizing compression.

Predictive quality and retrieval. Perplexity measures aggregate predictive fit; it does not localize memory operations. RULER includes needle-retrieval variations as well as more complex long-context tasks (Hsieh et al., 2024). We use a stated subset within the trained context length, not a full-suite score or evidence of length extrapolation. Zero-shot likelihood evaluation complements these probes by testing whether near-equal language-model loss transfers to downstream discrimination.

Table 1: Core R1 interventions. GDN QK/V include their short convolutions; update control excludes the output gate. All variants execute 12 layers and retain independent states/caches. Unique counts deduplicate parameter aliases.

Model	Tied parameter families	Unique (M)	Saved (%)
N0	None	403.363	0.00
N1	GDN QK	385.622	4.40
N2	GDN QK and update control	385.484	4.43
N3	GDN V	385.622	4.40
L1	GDN QK and update control; Full Q/K/V/O	338.298	16.13
L0	Whole GDN blocks; whole Full blocks (separate groups)	108.188	73.18

3 Addressing, Content, and Cross-Depth Sharing

3.1 The unchanged GDN update

For one head at depth ℓ, let $x_t^{(\ell)}$ be the normalized input, $q_t, k_t \in \mathbb{R}^{d_k}$ the unit-normalized query and key, $v_t \in \mathbb{R}^{d_v}$ the value, and $S_t \in \mathbb{R}^{d_k \times d_v}$ the memory. Suppressing the depth index, our implementation computes

$$q_t = \mathrm{norm}_2(\mathrm{SiLU}(C_Q(W_Q x)_t)), \qquad k_t = \mathrm{norm}_2(\mathrm{SiLU}(C_K(W_K x)_t)), \tag{1}$$

$$v_t = \mathrm{SiLU}(C_V(W_V x)_t), \qquad \alpha_t = \exp[-\exp(A)\,\mathrm{softplus}(W_a x_t + b_\Delta)], \tag{2}$$

$$\beta_t = \sigma(W_b x_t + b_b), \qquad \bar{S}_t = \alpha_t S_{t-1}, \tag{3}$$

$$S_t = \bar{S}_t + \beta_t k_t (v_t - \bar{S}_t^\top k_t)^\top, \qquad o_t = S_t^\top q_t / \sqrt{d_k}. \tag{4}$$

Here C_Q, C_K, C_V are causal depthwise short convolutions. Head outputs pass through per-head RMS normalization, an input-dependent SiLU output gate, and an output projection. The decay/write controls are distinct from this *output gate*. We retain the recurrence, kernels, and residual architecture; only parameter aliases change.

3.2 Interventions and matched controls

Write $\theta_A = (W_Q, W_K, C_Q, C_K)$ for addressing, $\theta_V = (W_V, C_V)$ for values, and $\theta_U = (W_a, W_b, b_b, A, b_\Delta)$ for decay/write control. N1 imposes $\theta_A^{(\ell)} = \theta_A$ across GDN layers. N2 also ties θ_U. N3 instead ties θ_V, leaving addressing and control independent. No configuration merges the states $S_t^{(\ell)}$ or the causal convolution histories.

L1 extends N2 by sharing the entire Full-attention mixer across Full layers, including Q/K/V and output projections. GDN value/output pathways, FFNs, and block norms remain independent. L0 shares whole blocks by mixer role: one GDN block and one Full block are each executed six times. Its FFNs and block norms are shared *within* each role, not as one FFN across both roles. Even in L0, every expanded depth has a separate cache/state entry. GDN and Full matrices are never directly tied to each other.

Exactly matching parameter reduction. Each of six GDN layers has input width 1536, total Q and K width 1152 each, total V width 2304, and convolution width four. Consequently,

$$P_{QK} = 2(1536 \cdot 1152 + 4 \cdot 1152) = 3{,}548{,}160, \tag{5}$$

$$P_V = 1536 \cdot 2304 + 4 \cdot 2304 = 3{,}548{,}160. \tag{6}$$

Either tie removes $5P_{QK} = 17{,}740{,}800$ parameters. N1 versus N3 therefore controls the *amount* of reduction exactly. It does not equalize optimization sensitivity or prove why one component tolerates tying better.

3.3 What the hypothesis does and does not say

The operational hypothesis is that a shared addressing rule can serve depth-specific content transformations with a small predictive penalty. It does not require $q_t^{(\ell)} = q_t^{(m)}$ or $k_t^{(\ell)} = k_t^{(m)}$: the

same function is applied to different inputs and convolution histories. Nor does it require an address direction to have the same semantic label at every depth.

The recurrence illustrates this distinction algebraically. For an orthogonal address-space transformation $R \in \mathbb{R}^{d_k \times d_k}$, set $\tilde{q} = Rq$, $\tilde{k} = Rk$, and $\tilde{S} = RS$. Then

$$\tilde{S}^\top \tilde{k} = S^\top k, \qquad \tilde{S}_t = R[\bar{S}_t + \beta_t k_t (v_t - \bar{S}_t^\top k_t)^\top], \qquad \tilde{S}_t^\top \tilde{q}_t = S_t^\top q_t. \tag{7}$$

The abstract update/readout is therefore covariant to orthogonal changes of address coordinates. This explanatory identity is not a new theorem or proof that independently trained layers can share weights. In particular, the nonlinear convolutional parameterization need not realize arbitrary rotated maps. Actual parameter tying is a stricter constraint that must be evaluated empirically.

4 Experimental Setup

Backbone. R1 alternates $[\mathrm{GDN}, \mathrm{Full}] \times 6$, starting with GDN. Hidden width is 1536; SwiGLU intermediate width is 3840; the 32,000-token embedding and LM head are tied. Each expanded layer applies a pre-normalized mixer residual and a pre-normalized FFN residual. GDN has nine heads with $d_k = 128, d_v = 256$; Full attention has 12 heads of width 128, causal global attention, and RoPE base 10,000. Context length is 4096. All core variants share this expanded architecture; they are internal controls, not reproductions of the original GDN-H1.

Data and budget. We use a frozen selection of 20 training Parquet files from FineWeb-Edu `sample-100BT` (Penedo et al., 2024), with a separate file reserved for validation. This is a subset of that collection, not a 100B-token training run. The tokenizer and packed manifest are fixed. A global batch contains 128 sequences with 4096 predicted tokens each, giving 524,288 prediction tokens per update. The primary checkpoint is exactly 19,500 updates, or 10,223,616,000 prediction tokens. The original 15B-token learning-rate schedule is retained: this is an intermediate-budget comparison, not an independently annealed 10B-token run.

Training uses AdamW, betas $(0.9, 0.95)$, weight decay 0.1, gradient clipping 1.0, peak learning rate 3×10^{-4}, 150M-token warmup, token-based cosine decay, BF16-mixed computation, and seed 3407. Physical GPU allocation and accumulation can differ; global batch is fixed. N0 and N3 at 19,500 were replayed from complete 19,000-step checkpoints, restoring optimizer and data-cursor state with changed physical layout. This is not claimed to be bitwise identical to their original trajectories. Source selection and budget are matched, but identical example order across every variant is not established.

Held-out language modeling. The online monitor uses the first 128 validation sequences. The primary evaluation skips these and evaluates the next 4096 sequences, totaling 16,777,216 prediction tokens. FP32 cross-entropy from BF16-forward logits is accumulated as $\exp(\sum_i \ell_i / N)$; batch perplexities are not averaged. All models use the same token window. File-level separation from training is established; additional document-level decontamination is not claimed. WikiText-2 raw test and LAMBADA complement this evaluation, but their metrics are not interchangeable with packed FineWeb-Edu token PPL.

Zero-shot downstream prediction. An independent adapter to `lm-eval` 0.4.9 evaluates PIQA, HellaSwag, WinoGrande, ARC-Easy/Challenge, Social IQa, BoolQ, OpenBookQA, LAMBADA, and WikiText-2. We use no instruction tuning, chat template, or few-shot examples. For candidate a and prompt p, the score is

$$\log p_\theta(a \mid p) = \sum_{j=1}^{|a|} \log p_\theta(a_j \mid p, a_{<j}), \tag{8}$$

excluding prompt tokens from the sum. Normalized accuracy follows the harness task's candidate-length normalization. LAMBADA additionally checks greedy token correctness across the target answer. Likelihood-based evaluation does not assume a base model can follow conversational instructions. Appendix B specifies metrics, counts, and implementation checks.

Table 2: Main evaluation at 19,500 updates (10.224B training tokens). FWE is fixed-window token PPL; Wiki is WikiText-2 raw test word PPL. LAMBADA reports answer PPL and accuracy (percent). All entries use complete audited evaluations.

Model	Unique (M)	Saved (%)	FWE PPL↓	Wiki PPL↓	LMB PPL↓	LMB acc↑
N0	403.363	0.00	10.974	26.664	39.581	32.64
N1	385.622	4.40	11.000	27.069	38.237	32.19
N2	385.484	4.43	10.993	27.175	40.460	31.44
N3	385.622	4.40	11.142	27.055	42.686	31.79
L1	338.298	16.13	11.199	27.742	45.306	30.25
L0	108.188	73.18	13.862	36.195	112.716	23.50

Table 3: Zero-shot accuracy (percent) at 19,500 steps. HellaSwag and ARC-C use normalized accuracy. The GDN-style average includes LAMBADA accuracy from Table 2 and the seven tasks below, equally weighted. These are single-training-seed point estimates.

Model	PIQA	Hella.	Wino.	ARC-E	ARC-C	SIQA	BoolQ	Avg.
N0	66.76	39.24	52.64	56.57	27.82	38.13	61.80	46.95
N1	66.00	38.42	51.22	58.67	26.88	37.77	61.74	46.61
N2	65.72	38.91	49.33	55.93	27.39	37.21	61.47	45.92
N3	65.89	38.12	50.91	56.06	25.94	38.69	60.61	46.00
L1	66.97	37.75	53.12	56.44	27.65	39.51	60.12	46.48
L0	62.51	32.38	49.72	51.47	25.43	37.67	61.71	43.05

Controlled retrieval. The pinned official RULER `niah_single_1` and `niah_multikey_2` configurations are evaluated at total context budgets 2048 and 4096, with 500 examples per condition and common seed 42. Both use word keys and numeric values, while differing in distractor structure. Prompts reserve 128 tokens for greedy generation. We use the official reference-string match metric. This is a two-task subset, not the complete 13-task benchmark (Hsieh et al., 2024), and it tests within-context retrieval rather than length extrapolation.

5 Results and Parameter Efficiency

5.1 Matched-budget language modeling

The fixed 16M-token evaluation orders the models N0, N2, N1, N3, L1, and L0 from lower to higher perplexity. N0 remains best on this window: our result is not an improvement claim. Address-only tying is close to N0, while the exactly parameter-matched value-only tie incurs a larger penalty. Extending addressing ties to update control changes few additional parameters and has a similarly small loss difference. Whole-block tying is much more aggressive, removing roughly three quarters of unique parameters but producing a substantially larger language-modeling penalty.

This distinguishes *how much* is tied from *which component* is tied. The N1/N3 difference cannot be explained solely by unequal parameter counts. It remains compatible with optimization differences, path sensitivity, and compensation by untied modules. Small loss differences from one training seed do not establish statistical equivalence. We use "small penalty" descriptively, not as a universal no-loss compression guarantee.

5.2 Downstream discrimination and retrieval

The eight-task average is 46.95% for N0, 46.61% for N1, 45.92% for N2, 46.00% for N3, 46.48% for L1, and 43.05% for L0. The ordering is not identical to FWE PPL: N2 has slightly better FWE PPL than N1 but a lower downstream average. N1 exceeds N0 on ARC-E and LAMBADA PPL, but not on the aggregate or LAMBADA accuracy. These task-specific changes do not establish a general advantage. OpenBookQA is separate to preserve the eight-task definition. Current point estimates use one trained seed; paired example-level uncertainty would address evaluation sampling, not training variability.

Table 4: Additional evaluation at the same 19,500-step checkpoint. RULER entries are official match percentages over 500 examples per condition. S is `niah_single_1`; MK is `niah_multikey_2`. All 12,000 model–example evaluations are complete and identity-checked. No full-suite RULER score is implied.

Model	OBQA acc_n ↑	S / 2K↑	S / 4K↑	MK / 2K↑	MK / 4K↑
N0	32.20	100.0	80.2	3.4	1.2
N1	32.40	98.8	37.6	3.8	0.4
N2	31.20	96.4	74.8	3.8	0.6
N3	34.40	99.6	70.2	0.4	0.4
L1	32.80	100.0	68.6	1.8	0.4
L0	28.80	99.2	55.0	0.6	0.2

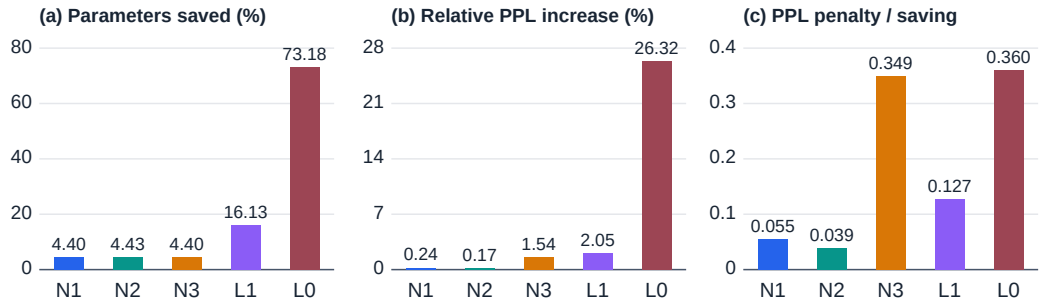


Figure 2: Observed trade-offs at the common 19,500-step checkpoint on the audited 16M-token window. Panels show parameter savings, relative PPL penalty, and their ratio. N1/N3 have identical savings but different penalties. L0's ratio must not conceal its much larger absolute penalty. These are single-seed point estimates, not equivalence tests. Source snapshot and vector-figure code are retained.

Retrieval exposes a limitation not visible from near-equal PPL. On single-needle retrieval at 2K, scores range from 96.4% to 100%. At 4K, N0 reaches 80.2%, N1 37.6%, N2 74.8%, and N3 70.2%. Thus N1 loses 42.6 percentage points despite its 0.24% relative FWE-PPL increase; preserving aggregate likelihood does not guarantee preserved retrieval. N2 is substantially closer to N0 on this probe, but one seed does not establish that control sharing causes the difference. L1 and L0 reach 68.6% and 55.0%. Multi-key scores are only 0.2–3.8% across conditions, a near-floor regime that cannot support functional equivalence. These are whole-hybrid-model behaviors, not localization of retrieval to GDN alone.

5.3 Separating parameter saving from predictive cost

For unique count P_i and mean held-out NLL L_i, define saving $s_i = 100(1 - P_i/P_0)$ and relative PPL change $d_i = 100[\exp(L_i - L_0) - 1]$. Alongside these, report

$$\eta_i = d_i/s_i, \qquad s_i > 0, \tag{9}$$

as a descriptive penalty per percentage point of parameter saving. Negative values indicate lower PPL than N0; positive values indicate a cost. N0's ratio is undefined. This is not a marginal derivative or a causal estimate of removing one parameter. An aggressively compressed model can have a favorable ratio despite an undesirable absolute loss, so all three quantities must be shown. The exact N1/N3 parameter match is a stronger control than normalizing unmatched experiments by this ratio.

We also track the normalized penalty at exactly matching online validation steps (Appendix D). This checks whether an ordering is confined to an endpoint. It is a diagnostic on a repeatedly monitored, smaller window, not an independent held-out set. Historical online values and separate FP32-loss evaluation must not be spliced into one trajectory.

6 Discussion and Limitations

Hypothesis: separating storage addressing from semantic retrieval. We hypothesize a functional distinction between *how to access a memory state* and *which information is relevant and how to represent it*. Under this interpretation, a shared GDN QK pathway supplies a reusable storage read/write rule, while depth-dependent hidden representations and untied pathways select and transform the information accessed through that rule. Full attention and FFNs are candidate contributors to the latter function, not components whose exclusive semantic role has been established. The hypothesis allows different contents at different depths under a common access mechanism; it does not require fixed semantic labels for memory coordinates.

This interpretation predicts that address-side tying can be more tolerable than comparably sized content-side tying, and that tying potential semantic-selection pathways may reveal different costs. The matched N1/N3 comparison bears on the first prediction. Testing the second requires matched-budget Full-QK and FFN controls and interventions that localize retrieval to particular pathways. Whole-model PPL or successful whole-model needle retrieval alone does not prove that these functions form independent modules. The present evidence motivates the hypothesis while leaving compensation and regularization as alternatives.

A common rule is not a common semantic label. Our claim concerns a learned operation on layer-dependent representations. A direction need not retain the same semantic interpretation at every depth. State contents and input representations can change while addressing parameters remain identical. This is the operational sense in which storage addressing and stored content can be separated. It does not establish that QK contains no semantics or that Full attention alone performs semantic retrieval.

Alternative explanations and missing isolations. Independent FFNs, GDN output paths, and Full-attention layers may absorb the cost of tying. QK and V may differ in optimization sensitivity even when their counts match. Shared parameters aggregate gradients from several depths and can regularize the model. The current interventions distinguish tolerances to tying, not a unique causal explanation. Full-only QK and FFN-only controls would strengthen localization, but partially evaluated lower-budget controls cannot substitute for common-budget evidence. They are excluded from the main tables.

Scale and uncertainty. The main study covers one 400M-class backbone, one seed, and one GDN:Full ratio. Completed 1.3B comparisons, a 3:1 ratio, and repeated seeds are not part of the main result. Identical seed labels do not eliminate differences from tying or execution layout. More shared layers can increase savings, but the total saving fraction is constrained by the tied component's share of the model, not unbounded growth with depth. These limits preclude a universal claim about all recurrent-memory architectures.

Systems implications. Tying reduces unique weights, gradients, and optimizer states. It does not remove expanded layers or their matrix multiplications: $X^{(\ell)}W_Q$ and $X^{(m)}W_Q$ must still be computed for different inputs. Recurrent states, convolution histories, and Full-attention KV caches remain separate. We claim parameter/storage efficiency, not demonstrated FLOP reduction, throughput improvement, or energy savings. Indirect gains from microbatch size, less recomputation, or weight-cache reuse require matched-hardware measurements.

7 Conclusion

We study cross-depth reuse of addressing parameters in gated recurrent memory. Selective QK tying preserves separate states and content paths; a parameter-matched V-only control isolates component choice from reduction amount. Common-checkpoint language modeling finds a smaller penalty for address-side than value-side tying in this setting, while whole-block looping yields a different, more costly compression trade-off. The retrieval probe also reveals a substantial QK-only deficit hidden by near-equal PPL, so addressing reuse is not a universally lossless constraint. Storage–semantic retrieval separation remains a hypothesis rather than an established decomposition. Its limits must be assessed across behaviors, scales, seeds, and compensating pathways.

REPRODUCIBILITY STATEMENT

We specify architecture, tying groups, data selection, token accounting, optimizer, checkpoint selection, and evaluation. Dataset/task revisions, checkpoint identities, per-example predictions, and code hashes are retained. Appendix B describes scoring checks and Appendix C documents replay caveats. All reported figures and tables use frozen evaluation snapshots rather than mutable training-log endpoints.

AI USE STATEMENT

Generative AI tools assisted with research framing, experimental planning, code implementation and debugging, literature discovery, mathematical exposition, interpretation, and drafting and editing this manuscript. AI outputs are not independent evidence. The authors are responsible for verifying submitted claims, references, numerical results, and code and take responsibility for the final content.

REFERENCES

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.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions, 2019. URL https://arxiv.org/abs/1905.10044.

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.

Mostafa Dehghani, Stephan Gouws, Oriol Vinyals, Jakob Uszkoreit, and Lukasz Kaiser. Universal transformers, 2018. URL https://arxiv.org/abs/1807.03819.

Chunyuan Deng, Yizhe Zhang, Rui-Jie Zhu, Yuanyuan Xu, Jiarui Liu, T. S. Eugene Ng, and Hanjie Chen. LT2: Linear-time looped transformers, 2026. URL https://arxiv.org/abs/2605.20670.

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.

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.

Kimi Team. Kimi linear: An expressive, efficient attention architecture, 2025. URL https://arxiv.org/abs/2510.26692.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. ALBERT: A lite BERT for self-supervised learning of language representations, 2019. URL https://arxiv.org/abs/1909.11942.

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

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering, 2018. URL https://arxiv.org/abs/1809.02789.

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.

Telmo Pessoa Pires, António V. Lopes, Yannick Assogba, and Hendra Setiawan. One wide feedforward is all you need. In *Proceedings of the Eighth Conference on Machine Translation*, 2023. URL https://aclanthology.org/2023.wmt-1.98/.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. WinoGrande: An adversarial winograd schema challenge at scale, 2019. URL https://arxiv.org/abs/1907.10641.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. SocialIQA: Commonsense reasoning about social interactions, 2019. URL https://arxiv.org/abs/1904.09728.

Songlin Yang, Jan Kautz, and Ali Hatamizadeh. Gated delta networks: Improving mamba2 with delta rule. *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2412.06464.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence?, 2019. URL https://arxiv.org/abs/1905.07830.

A Configuration and Parameter Accounting

Setting	400M R1 comparison
Expanded layers / order	12 / $[\mathrm{GDN}, \mathrm{Full}] \times 6$
Hidden / FFN widths	1536 / 3840
Vocabulary / LM head	32,000 / tied with embedding
GDN heads / key / value dimensions	9 / 128 / 256
Q/K/V convolution	Depthwise causal, width 4, no bias
Full attention	12 heads, global causal, RoPE base 10,000
Normalization	Pre-RMSNorm; epsilon 10^{-5}
Context / global batch	4096 predictions / 128 sequences
Prediction tokens per step	524,288
Main checkpoint	19,500 / 10,223,616,000 tokens
Optimizer	AdamW $(0.9, 0.95)$, weight decay 0.1
Peak LR / warmup	3×10^{-4} / 150M tokens
Schedule endpoint	15B tokens, unchanged
Precision / seed / gradient clipping	BF16 mixed / 3407 / global norm 1.0

Table 5: Shared recipe. Normalization and one-dimensional control parameters are excluded from weight decay according to the optimizer grouping.

The FineWeb-Edu revision is fc9850dff5e2d0f8f776efe41b24a1c49556cfc5. The selected 20 training files, validation-file identity, tokenizer checksum, and packed manifest identify the actual corpus; the collection name alone is insufficient. Each example reads 4097 IDs to form 4096 input/next-token pairs. The extra ID is not an additional prediction. A nominal 15B-token phase contains $\lfloor 15 \times 10^9 / 524288 \rfloor = 28{,}610$ complete updates.

For a component of size p repeated m times, global tying removes $(m-1)p$ unique parameters. With g independent sharing groups, the saving is $(m-g)p$ for equal-sized copies. Given other parameters P_{rest}, the global-tie fraction is $(m-1)p/(P_{\mathrm{rest}} + mp)$. Unique counts deduplicate aliases and count the embedding/head once; operation counts do not deduplicate depth. In R1, one GDN control family contains $2 \cdot 1536 \cdot 9 + 3 \cdot 9 = 27{,}675$ parameters. Tying six copies removes another 138,375 beyond N1 to obtain N2. This excludes the 1536×2304 output gate.

B Evaluation Definitions and Verification

The task sources are PIQA (Bisk et al., 2019), HellaSwag (Zellers et al., 2019), WinoGrande (Sakaguchi et al., 2019), ARC (Clark et al., 2018), Social IQa (Sap et al., 2019), BoolQ (Clark et al., 2019), OpenBookQA (Mihaylov et al., 2018), LAMBADA (Paperno et al., 2016), and WikiText (Merity et al., 2016). We distinguish these dataset sources from the concrete harness versions and splits used below.

Task	Split	Examples	Metric
PIQA	Validation	1,838	Accuracy
HellaSwag	Validation	10,042	Normalized accuracy
WinoGrande	Validation	1,267	Accuracy
ARC-Easy	Test	2,376	Accuracy
ARC-Challenge	Test	1,172	Normalized accuracy
Social IQa	Validation	1,954	Accuracy
BoolQ	Validation	3,270	Accuracy
OpenBookQA	Test	500	Normalized accuracy
LAMBADA OpenAI	Test	5,153	Target PPL and accuracy
WikiText-2 raw	Test	62 documents	Word perplexity
RULER S / MK	Generated	500 per condition	Official string-match score

Table 6: Task identities and counts. Few-shot count is zero. WikiText-103 is not evaluated. RULER has two lengths per task.

Candidate scoring and checks. Context and continuation are tokenized consistently at their boundary, including the harness handling of trailing context whitespace. Only continuation positions contribute to Eq. (8); the prompt remains visible. Multi-token answers are scored by sums of conditional log probabilities, not just the first token. The harness applies task-defined normalization and selects the highest candidate score. Rolling likelihood counts target positions according to that protocol. Packed token PPL and word-normalized WikiText PPL are not pooled.

Unit tests cover candidate masking, multi-token continuations, and adapter behavior. Actual-checkpoint BF16-forward/FP32-log-softmax scores were compared with independently computed cross-entropy. A two-example PIQA smoke test checked execution only and is excluded from formal scores. Formal files have no sample limit and record checkpoint identity. Prepared caches were reused offline; worker-specific cache locations do not define different samples.

RULER generation and merging. The official generator/scorer revision is `c3f5e3b4f87f97e048793bb510a3a6b19a46bf3a`. Both configurations use word keys and numeric values; MK includes distractor key/value entries. The prompt contains the official answer prefix but not the answer. Greedy generation uses no chat format or extra BOS/EOS, stopping at EOS or 128 new tokens. Context budgets include output reservation; coarse generator length adjustment can yield shorter prompts. Overflow is rejected rather than truncating the needle.

Each expanded layer has a separate recurrent/convolution state or attention KV cache, reset between examples. CPU tests compare cached and full-prefix generation. All six trained checkpoints passed comparisons of the first 16 generated tokens on one sample from each of four task/length conditions. This is a targeted parity check, not exhaustive equality over all prompts. Official scoring checks reference strings within generated text case-insensitively, not strict whole-answer exact match. Merging requires every sample exactly once, complete shard markers, and matching dataset/checkpoint identities. Partial shard averages are not final scores.

Averages and uncertainty. The eight-task average includes LAMBADA accuracy, PIQA accuracy, HellaSwag normalized accuracy, WinoGrande accuracy, ARC-E accuracy, ARC-C normalized accuracy, SIQA accuracy, and BoolQ accuracy, equally weighted by task. OBQA is separate; PPL is never included in an accuracy average. Point estimates do not include training-seed uncertainty. Future paired bootstrap intervals should reuse per-example outputs and be distinguished from seed variation.

C Checkpoint Selection and Replay

The common step aligns available core models under a deadline; it is not each model's validation-optimal checkpoint. N1/N2/L0/L1 retained exact 19,500-step checkpoints. N0/N3 were replayed from complete 19,000-step archives into separate run directories, preserving higher-step originals. Optimizer and data-source state were restored, while GPU layout changed. Replayed results are not claimed to be the missing bitwise-identical original checkpoints. Every evaluation uses independently pinned weights, not a mutable latest-file pointer. Checkpoint identities and separate logs prevent a score from silently following a replaced checkpoint.

Equal seed and nominal token count do not prove equal floating-point trajectories or identical example ordering. Small differences are interpreted conservatively. Matched-order repeated seeds are required to estimate training variability and demonstrate a reproducible advantage.

D Historical Training-Curve Diagnostic

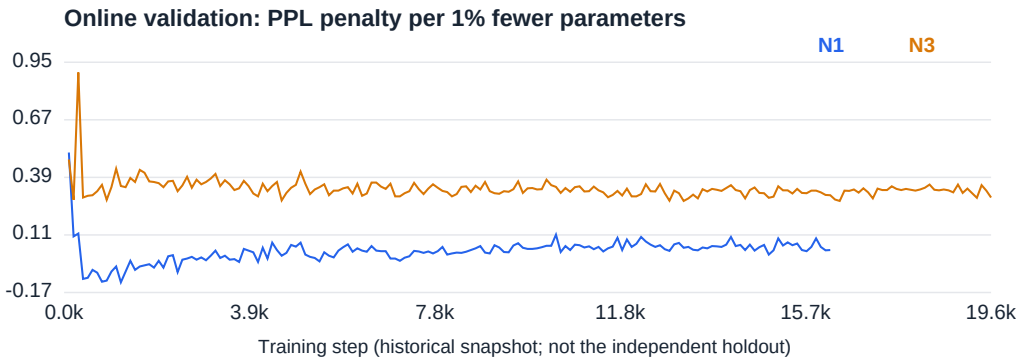


Figure 3: Historical online-monitor snapshot for parameter-matched N1 and N3, relative to N0 at matching logged steps. Frozen on September 24, it ends at available steps and does not extrapolate to 19,500. This repeatedly monitored window and numerical path differ from the independent 16M-token evaluation in Figure 2. Lines connect observations without smoothing and do not create intermediate measurements.

An early negative penalty means lower recorded PPL than N0 at that step, not necessarily a durable advantage. Optimization noise, initialization under tying, regularization, and finite-window variation are alternative explanations. The trajectory identifies persistent versus transient ordering; the common-window evaluation remains the primary language-model comparison.

E Scope of the Available Evidence

Tables 2–4 contain complete common-checkpoint evaluations of six models, including 12,000 RULER model–example pairs. Additional training seeds, completed 1.3B/3:1 comparisons, matched-budget Full-QK/FFN controls, and matched-hardware throughput remain outside the evidence. The retrieval results do not localize the cause of N1's 4K deficit. Replication and pathway-specific interventions are needed before making a causal claim about update-control sharing or semantic specialization.