M^2RNN: Non-Linear RNNs with Matrix-Valued States for Scalable Language Modeling

Research paper discussion

Paper authors: Mayank Mishra, Shawn Tan, Ion Stoica,
Joseph E. Gonzalez, Tri Dao

arXiv:2603.14360v2 | Revised 13 May 2026
Reading: 9 October 2026

Large nonlinear memory becomes practical when used sparingly.

Three bottlenecks motivate one architecture

State computation	Memory capacity	Hardware efficiency
Nonlinear transitions can implement rich step-by-step updates.	Outer-product writes expand a vector state into a matrix.	Matrix states provide useful GEMM tiles at small batch sizes.
Linear recurrences with restricted transitions have expressivity limits.	A small vector state can be a serious language-modeling bottleneck.	Time steps remain sequential; sparse use controls the cost.

The strongest practical result: add a few nonlinear layers to an existing hybrid.

Paper, Secs. 1–3, PDF pp. 1–10.

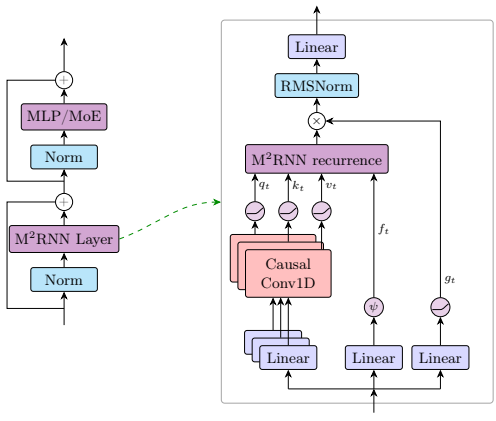
Write, transform, retain, and read a matrix state

$$\mathbf{Z}_t = \tanh(\mathbf{H}_{t-1}\mathbf{W} + k_t v_t^\top),$$
$$\mathbf{H}_t = f_t \mathbf{H}_{t-1} + (1 - f_t)\mathbf{Z}_t,$$
$$y_t = \mathbf{H}_t^\top q_t + w_r \odot v_t.$$

- $\mathbf{H}_t, \mathbf{Z}_t \in \mathbb{R}^{K \times V}$; $\mathbf{W} \in \mathbb{R}^{V \times V}$; $\mathbf{H}_0 = 0$.
- $q_t, k_t \in \mathbb{R}^K$, $v_t, w_r \in \mathbb{R}^V$; $f_t \in [0, 1]$ is scalar per head.
- $\mathbf{W}$ mixes value channels within each row; q_t combines rows at readout.

Paper, Eqs. (10)–(12), PDF p. 7.

Most of the block can be computed before recurrence



Paper, Fig. 2 (cropped) and Eqs. (14)–(23), PDF p. 8.

- q, k, v: linear projections, causal Conv1D (width 4), then SiLU.
- Forget and output gates depend on layer inputs.
- After readout: multiply by g_t, apply RMSNorm, then project out.
- Pre-norm residual blocks surround the mixer and MLP/MoE.

Shared queries and keys buy more recurrent capacity

Multi-value formulation

$N_q = N_k = 1$, while $N_v = N_f = N_w = N$. Each value head retains its own $K \times V$ state.

$$K = 64, \quad V = 16 \quad \Rightarrow \quad \underbrace{KV = 1{,}024}_{\text{state elements/head}}, \qquad \underbrace{V^2 = 256}_{\text{transition parameters/head}}.$$

$$P_{\mathrm{MH}} \approx N(2K + 3V)d_m, \qquad P_{\mathrm{MV}} \approx 3NVd_m.$$

- Reuse q/k projections to allocate a larger state at a fixed projection budget.
- These are approximate *projection* counts; recurrent matrices and other parameters are excluded.

Paper, Algorithm 1 and Appendix A / Table 8, PDF pp. 11, 29.

The gate preserves a direct path through time

$$a_{t,n} = (\mathbf{W}_f x_t)_n, \qquad f_{t,n} = \exp[-\alpha_n \,\mathrm{softplus}(a_{t,n} + \beta_n)]\,.$$

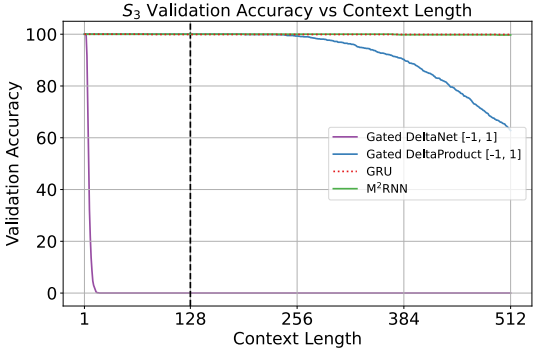
- α_n controls decay; β_n shifts the response. Use $\alpha_n \geq 0$ to keep $f \in [0, 1]$.
- $f_t = 1$: retain the state. $f_t = 0$: use the nonlinear candidate, which still depends on history.
- Initialize $\mathbf{W} = \mathbf{I}$ and clip recurrent-state gradients during BPTT.

Why input-only gating matters

Gate values can be precomputed across time. The nonlinear state transition still needs sequential evaluation.

Paper, Sec. 3.1, Eqs. (13), (17), PDF pp. 7–9. Interpretation of gate endpoints follows the recurrence.

Nonlinear models generalize on the tested S_3 task



- Train at length 128; evaluate to 512.
- M^{2}RNN and GRU: reported accuracy $\geq 99.5\%$.
- DeltaProduct loses accuracy as length grows.

The recurrence can embed a vector RNN in one row. The experiment tests S_3, whereas the complexity motivation discusses S_5.

Paper, Theorem 1 / Fig. 3, PDF p. 9; proof in Appendix B, p. 30.

Efficient tiles do not remove the sequential dependency

Forward

Parallelize over batch and heads. Keep **H** and **W** on chip; scan time sequentially.

GEMM shape: $(K, V) \times (V, V)$.

Unlike batch-based vector tiles, this shape does not require padding a small batch to 16.

Backward

Recompute the trajectory, cache states in HBM, then run the reverse sweep.

Temporary state storage: $O(BTNKV)$.

The current Triton backward kernel is memory-bandwidth bound.

Core recurrent work: $O(BTNKV^2)$; linear in length at fixed state dimensions.

Paper, Sec. 3.4 and Secs. 4.1–4.2 / Algorithms 1–2, PDF pp. 10–12.

Two model scales share a 100B-token training protocol

	Dense	Mixture of Experts
Parameters	410M scale	7B total, 1.1B active
Layers	24	40
Training GPUs	32 H100s	64 H100s

- Nemotron-CC-v2 high-quality subset; identical data and order.
- 4,096-token training sequences; global batch of 1,024 sequences.
- AdamW, peak learning rate 3×10^{-4}; BF16 with FP32 gradient accumulation.
- Transformer++ uses RoPE; linear recurrent and hybrid models use NoPE.

Paper, Sec. 5 / 5.2, PDF p. 15. Table captions round active parameters to 1B.

The three-way hybrid changes only a few mixer layers

Design	Attention layers	Recurrent layers
Homogeneous	None	One recurrent family
Two-way hybrid	1 in every 8	GDN, Mamba-2, or M^2RNN
Three-way hybrid	1 in every 8	GDN/Mamba-2 plus M^2RNN

- **+1**: replace the first recurrent layer.
- **+3 at 410M / +5 at 7B**: replace each layer immediately before attention.
- All variants are trained from scratch. Layer count and layer placement both change across these comparisons.

Paper, Sec. 5.1, PDF p. 15. GDN = Gated DeltaNet.

Pure M^{2}RNN reaches competitive language modeling

	410M		7B MoE	
Model	Wiki PPL ↓	Avg. acc. ↑	Wiki PPL ↓	Avg. acc. ↑
RNN	33.74	48.90	17.65	60.58
GRU	25.80	53.72	14.80	64.44
Mamba-2	22.93	54.30	13.73	65.22
GDN	22.70	53.99	13.89	64.28
M^{2}RNN	22.92	54.24	13.80	65.17

- M^{2}RNN closes much of the gap between vector RNNs and modern recurrent models.
- It remains slightly behind the best recurrent baseline on these metrics.

Paper, Tables 1–2, PDF p. 16. Average over ten 0-shot downstream accuracy columns.

One nonlinear layer is already useful at 7B

7B MoE model	Wiki PPL ↓	LAMBADA PPL ↓	Avg. acc. ↑
Transformer++	14.94	17.26	61.76
Hybrid Mamba-2	13.10	11.05	65.03
Hybrid GDN	13.51	11.06	64.64
Hybrid M^2RNN	13.00	10.58	65.56
Hybrid GDN +1	13.07	10.33	**65.80**
Hybrid GDN +5	**12.85**	**10.29**	65.66

Single-layer change relative to Hybrid GDN

WikiText: -0.44 PPL. Downstream average: $+1.16$ percentage points.

Paper, Table 2, PDF p. 16. +1 / +5 denote the number of substituted M^2RNN layers.

At 410M, gains depend on the metric

410M model	Wiki PPL ↓	LAMBADA PPL ↓	Avg. acc. ↑
Hybrid GDN	21.89	29.02	54.04
Hybrid M^{2}RNN	21.53	37.31	54.15
Hybrid GDN +1	21.39	33.78	53.75
Hybrid GDN +3	**21.26**	**28.93**	**55.23**

- One layer improves WikiText by 0.50 PPL, but worsens LAMBADA PPL and the downstream average.
- Three layers improve all three reported aggregate metrics in this comparison.

Paper, Table 1, PDF p. 16. This qualifies the prose claim of gains across all benchmarks.

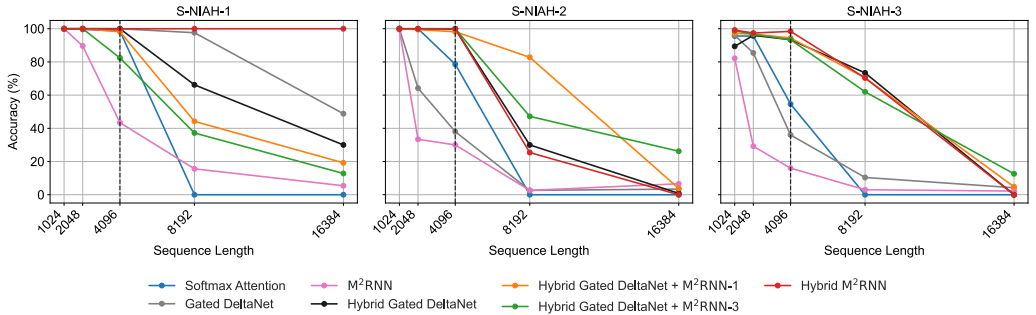
Attention hybrids outperform pure recurrent retrieval

Model	410M mean score ↑	7B mean score ↑
Pure GDN	30.8	42.2
Pure M^{2}RNN	26.9	37.9
Transformer++	34.6	45.5
Hybrid GDN	43.1	51.4
Hybrid M^{2}RNN	**46.9**	**55.6**
Hybrid GDN +1	45.5	54.2

- Means cover SQuAD, NQ, DROP, TriviaQA, FDA, and SWDE.
- Pure M^{2}RNN still trails GDN; the paper reports a roughly 3× smaller recurrent state for M^{2}RNN.

Paper, Tables 3–4 / Sec. 5.4.2, PDF p. 20. Scores follow each task's reported evaluation.

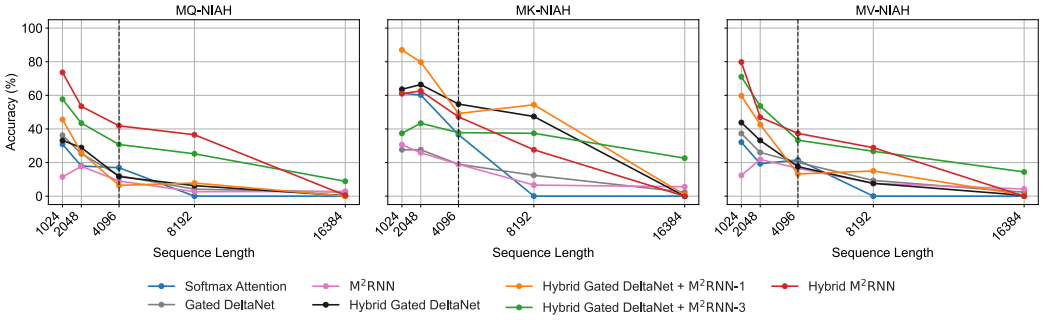
Single-needle retrieval shows distinct failure modes



At 410M, the +3 GDN hybrid retains more S-NIAH-2 accuracy at 16K. On S-NIAH-1, replacing GDN layers can reduce accuracy.

Paper, Fig. 6 top row, PDF p. 18; original legend retained. Dashed line: 4,096-token training length.

Multiple associations favor more nonlinear layers



At 16K, +3 has the highest accuracy in these 410M MQ/MK/MV panels. Absolute accuracy remains limited. The Mamba backbone also needs multiple layers for more reliable gains.

Paper, Fig. 6 bottom row and discussion of Figs. 7–9, PDF pp. 17–19. MQ: multiple queries; MK: distracting keys; MV: multiple values.

LongBench gains are large, but architecture-specific

	410M		7B MoE	
Model	Score ↑	Gain	Score ↑	Gain
Hybrid GDN	15.4	—	23.1	—
Hybrid M^{2}RNN	15.4	0.0	24.6	+1.5
Hybrid GDN +1	19.5	+4.1	30.4	+7.3
Hybrid GDN +3 / +5	23.0	+7.6	31.2	+8.1

- Eight selected tasks: summarization, few-shot learning, and code completion.
- 7B Hybrid Mamba-2: 18.8 → 13.1 with one layer; 22.7 with five.
- 7B GDN +1: MultiNews drops 16.7 → 6.0, despite the higher mean.

Paper, Tables 5–6, PDF p. 21. Gains computed from displayed means; scores are not a common accuracy metric.

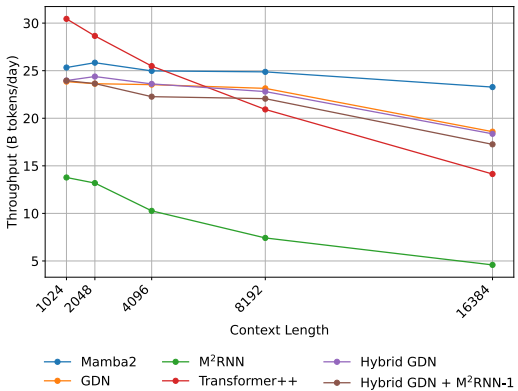
Capacity is a compelling explanation, with a confound

Model	Params	State size	Wiki PPL ↓	LAMBADA PPL ↓
RNN	406M	1,360	33.74	317.38
GRU	474M	1,360	25.80	63.83
M^2RNN	410M	86,016	22.92	33.63

- About 63.25× as many state elements as the vector baselines, at a parameter count close to RNN.
- GRU improves substantially at the same state size: gating matters too.
- Matrix state, read/write structure, and gate design change together. Table 7 does not isolate their individual effects.

Paper, Table 7 / Sec. 5.6, PDF pp. 21–22. State-size ratio is calculated from the table.

Sparse use keeps training throughput close to GDN



- 7B MoE, 8 HGX H100 GPUs, NVLink.
- Pure M^2RNN is substantially slower.
- The paper reports a throughput loss within 6% at 16K for Hybrid GDN $+1$.

Paper, Fig. 10 / Sec. 5.7, PDF p. 22. Training throughput in billions of tokens/day; no $+5$ or decoding-throughput curve.

Tensor parallelism trades topology dependence for sync

Topology-aware

One query/key group per TP GPU.

- $N_q = N_k = N_{TP}$.
- Local normalization and local recurrent heads.
- No added communication beyond standard TP.
- Projection sizes depend on the training topology.

Topology-independent

Share one query/key group globally.

- $N_q = N_k = 1$.
- Shard normalization; synchronize its statistics.
- Extra AllReduces: 1 forward, 3 backward.
- Parameter count stays independent of TP size.

Paper, Sec. 4.3 / Figs. 4–5, PDF pp. 12–14; Appendix C, pp. 30–31.

Three qualifications change how to use the result

1. **Complexity separation is conditional.** The TC^0 versus NC^1 separation uses a conjecture and specific computational assumptions. The direct experiment is S_3.
2. **Capacity and architecture are coupled.** The ablation changes more than state size; single- and multi-layer hybrids also use different layer locations.
3. **Implementation matters.** Algorithm 2 omits future-state gradient accumulation and has an incorrect state-gradient update. The chain-rule form is in the backup slides.

Paper, Secs. 2, 5.1, 5.6 and Algorithm 2, PDF pp. 3–5, 12, 15, 21–22. Complexity qualification: Merrill et al., ICML 2024, Sec. 3. Pseudocode issue does not establish a bug in the training code.

The strongest takeaway is a modular architecture result

A practical result at 7B

Hybrid GDN + one M^{2}RNN layer: -0.44 WikiText PPL, $+7.3$ LongBench points, and reported training-throughput loss within 6% at 16K.

- Matrix states make nonlinear recurrence competitive in language modeling.
- A few expensive recurrent layers can complement attention and efficient linear recurrence.
- The best layer count depends on backbone, task, and training cost.

Mishra et al., *M^{2}RNN*, arXiv:2603.14360v2

Key evidence: Tables 2 and 6, Fig. 10, PDF pp. 16, 21–22. Figures in this deck are reproduced or cropped from this version.

Backup: embedding a vector RNN in one matrix row

Set $f_t = 0$, $w_r = 0$, and $q_t = k_t = e_1$. Starting from $\mathbf{H}_0 = 0$:

$$\mathbf{H}_{t,1,:} = \tanh(\mathbf{H}_{t-1,1,:}\mathbf{W} + v_t^\top), \qquad \mathbf{H}_{t,i,:} = 0 \ (i > 1).$$

With $h_t = \mathbf{H}_{t,1,:}^\top$,

$$h_t = \tanh(\mathbf{W}^\top h_{t-1} + v_t), \qquad y_t = h_t.$$

- This embeds the vector recurrence in the abstract matrix recurrence.
- The proof allows the endpoint $f_t = 0$; the finite-parameter positive gate typically approaches that endpoint only in a limit.
- Output gating and RMSNorm are outside this core simulation argument.

Paper, Appendix B / Eq. (25), PDF p. 30; endpoint qualification follows Eq. (13), p. 7.

Backup: the recurrent gradient needs both paths

For $\bar{y}_t = \mathbf{H}_t^\top q_t$, let $u_t = \partial \mathcal{L} / \partial \bar{y}_t$ and $\mathbf{C}_t$ be the gradient from future time steps.

$$\mathbf{R}_t = \mathbf{C}_t + q_t u_t^\top,$$
$$\mathbf{D}_t = ((1 - f_t)\mathbf{R}_t) \odot (1 - \mathbf{Z}_t \odot \mathbf{Z}_t).$$

$$\boxed{\mathbf{C}_{t-1} = f_t \mathbf{R}_t + \mathbf{D}_t \mathbf{W}^\top.}$$

$$\partial_{f_t} \mathcal{L} = \langle \mathbf{R}_t, \mathbf{H}_{t-1} - \mathbf{Z}_t \rangle_F, \qquad \partial_{\mathbf{W}} \mathcal{L} = \sum_t \mathbf{H}_{t-1}^\top \mathbf{D}_t.$$

The printed Algorithm 2 omits $\mathbf{C}_t$ and overwrites the direct gradient branch with an incorrect product. Recomputing the trajectory also requires updating the local state each step.

Reading analysis derived from Eqs. (10)–(12), PDF p. 7; compare Algorithm 2, p. 12. The value residual and outer block contribute additional gradients.

Backup: projection sharing explains the state budget

Pattern	Approximate projection count	State at budget P
Multi-head	$N(2K + 3V)d_m$	$\frac{PKV}{(2K + 3V)d_m}$
Multi-query/key	$N(K + 2V)d_m$	$\frac{PKV}{(K + 2V)d_m}$
Multi-value	$3NVd_m$	$\frac{PK}{3d_m}$

- At $K = 64, V = 16$, these approximations give $S_{\mathrm{MV}}/S_{\mathrm{MH}} \approx 3.67$.
- Multi-value's listed projections sum to $(2K + 3NV + N)d_m$ before approximation.
- Recurrent matrices add NV^2 parameters, outside the table's projection count.

Paper, Appendix A / Table 8, PDF p. 29. Ratios and unapproximated sum calculated from the listed terms.

Backup: RMSNormTP synchronizes per-token statistics

For a feature shard D_n on GPU n:

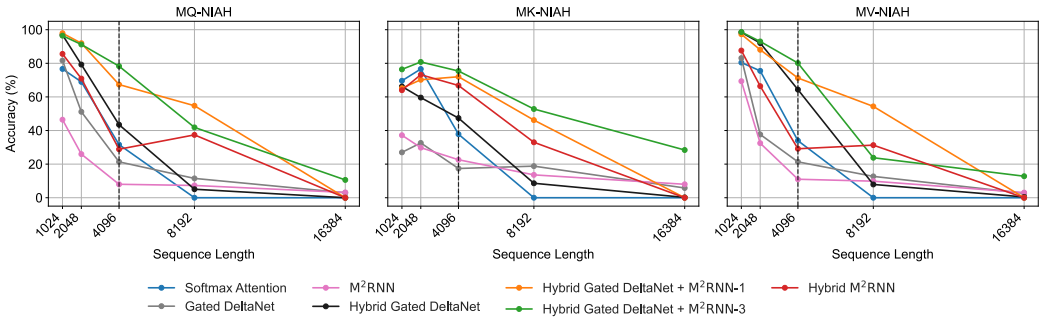
$$s = \left[\frac{1}{d} \sum_{n=1}^{N_{TP}} \sum_{j \in D_n} x_j^2 \right]^{-1/2}, \quad y_i = w_i x_i s.$$

$$p_i = w_i \nabla_{y_i} \mathcal{L}, \qquad r = \sum_{n=1}^{N_{TP}} \sum_{j \in D_n} p_j x_j, \qquad \nabla_{x_i} \mathcal{L} = s p_i - \frac{r s^3 x_i}{d}.$$

- AllReduce the squared-norm sum in forward and the inner product in backward.
- Each statistic is scalar *per token*; a batch of sequences has BT such statistics.
- Two further backward AllReduces aggregate the shared query/key activation gradients.

Paper, Appendix C / Eqs. (26)–(33), PDF pp. 30–31; Sec. 4.3.2, p. 14. Epsilon omitted as in the paper.

Backup: 7B benefits on multi-association retrieval



Original figure labels the multi-layer curve "+3"; Sec. 5.1 and the 7B tables specify +5. The figure is retained without relabeling.

Paper, Fig. 8 bottom row, PDF p. 19; original legend retained. Dashed line: 4,096-token training length.

Backup: references and author-provided resources

- Mayank Mishra, Shawn Tan, Ion Stoica, Joseph E. Gonzalez, Tri Dao.
  M^2RNN: Non-Linear RNNs with Matrix-Valued States for Scalable Language Modeling.
  arXiv:2603.14360v2, revised 13 May 2026.

- William Merrill, Jackson Petty, Ashish Sabharwal.
  The Illusion of State in State-Space Models. ICML 2024.
  Used to clarify the complexity-theoretic assumptions.

- Paper-provided resources:
  lm-engine training framework Model and kernel repository
  M^2RNN model collection