RAM-Net: Linear-Time Sequence Modeling with Sparsely Addressable State

Paper discussion | 3 November 2026

Paper authors: Kaicheng Xiao, Haotian Li, Liran Dong, Guoliang Xing

arXiv:2602.11958v2 • Accepted at NeurIPS 2026 (official list)

Official public implementation

Takeaway: sparse slot addressing improves fine-grained recall with low per-token state access.

The question: can a fixed state support precise recall?

The tension

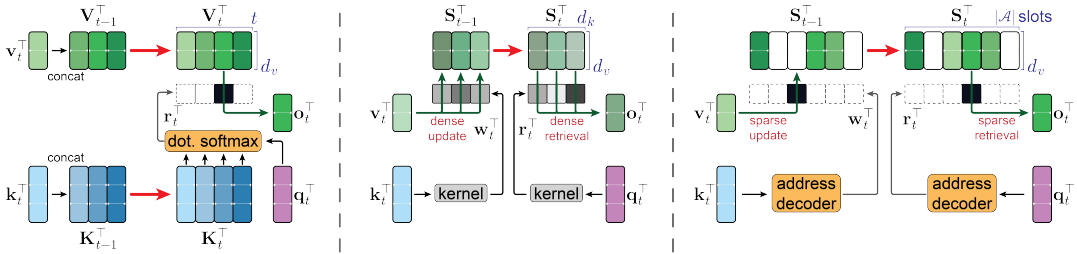
Full attention keeps separable token entries; recurrent linear attention compresses them into a shared fixed-size state.

- Shared writes superpose unrelated values.
- Dense additive reads bring interference into the output.
- Enlarging a densely accessed state raises per-step access cost.

RAM-Net's proposal: keep a bounded state, partition it into slots, and sparsify both writes and reads.

Paper, Secs. 1–2, PDF pp. 1–3.

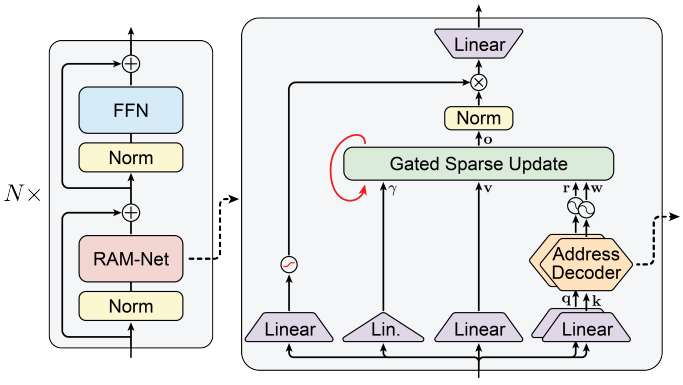
RAM-Net changes the unit of storage and access



- Full attention: per-token KV entries; score all cached keys.
- Linear attention: shared state; dense update and read.
- RAM-Net: fixed slot array; only selected slots are touched.

Original paper Fig. 1, PDF p. 3.

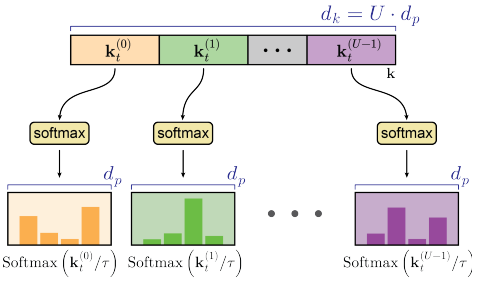
One layer separates address, update, and readout



Cropped from original paper Fig. 2, PDF p. 4; Sec. 3.

1. Project x_t to k_t, q_t, v_t, γ_t.
2. Decode sparse write/read addresses.
3. Update addressed slots with GSU.
4. Read, normalize, gate, and project.

Product Softmax addresses d_p^U slots with Ud_p logits



(a) Sub-vector Softmax

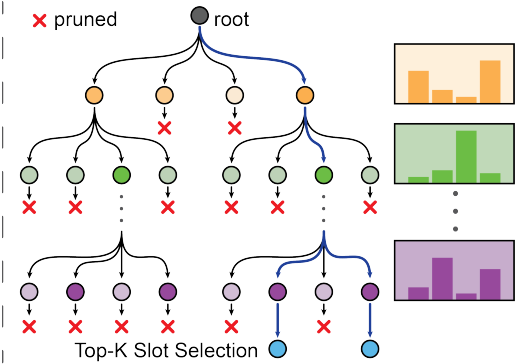
$$a = \sum_{u=0}^{U-1} a^{(u)} d_p^u, \qquad p^{(u)} = \mathrm{softmax}(k_t^{(u)}/\tau).$$

$$\rho_U(k_t)[a] = \prod_{u=0}^{U-1} p^{(u)}[a^{(u)}].$$

- $M = |\mathcal{A}| = d_p^U$ slots per head.
- $d_k = Ud_p$ key/query dimensions.
- Example:
  $U = 5, d_p = 4 \Rightarrow M = 1024,\ d_k = 20.$

Paper, Sec. 3.2, Eq. (4), PDF pp. 4–5; image cropped from Fig. 3.

Exact Top-K search never builds the full address vector



(c) Top-K Prefix Tree Search

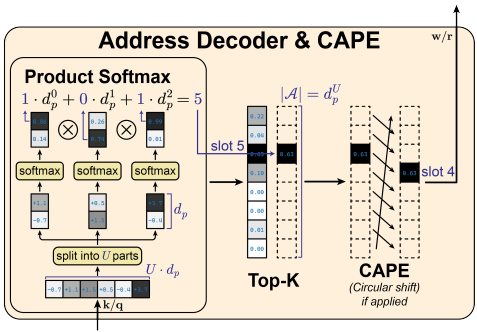
- Retain the best K prefixes at each digit.
- Extend each by d_p choices, then keep the best K.
- All prefixes share the same remaining digit distributions: a pruned prefix cannot return to the final Top-K.

decode cost $= O(UKd_p + Ud_p \log d_p)$.

Computation uses log probabilities; retained weights are **not renormalized**.

Paper, Secs. 3.2 and 4, Eq. (5), PDF pp. 5–6; image cropped from Fig. 3.

CAPE makes address overlap depend on relative position



Paper, Sec. 3.3, Eq. (6), PDF p. 5; cropped from Fig. 2.

$$\pi_t(a) = (a - t) \bmod M.$$

$$w_t[\pi_t(a)] = \mathcal{D}_{K,U}(k_t)[a],$$

$$r_t[\pi_t(a)] = \mathcal{D}_{K,U}(q_t)[a].$$

- A write at s and read at t overlap according to content *and* distance $t - s$.
- Some heads use CAPE; others use content-only NoPE.

GSU updates only addressed slots

For each slot a, keep content $s_t[a] \in \mathbb{R}^{d_v}$ and write mass $m_t[a]$:

$$\begin{aligned}
\lambda_t[a] &= \sigma(\mathrm{logit}(1 - w_t[a]) - \gamma_t)\,, \\
m_t[a] &= \lambda_t[a] m_{t-1}[a] + w_t[a], \\
\eta_t[a] &= \frac{w_t[a]}{m_t[a] + \epsilon}, \\
s_t[a] &= s_{t-1}[a] + \eta_t[a](v_t - s_{t-1}[a]).
\end{aligned}$$

$w_t[a]$: write weight; γ_t: input-dependent forgetting strength; $\eta_t[a]$: adaptive update fraction.

Paper, Sec. 3.4, Eqs. (7)–(8), PDF p. 6.

The sparse update has three useful consequences

1. **Exact inactivity:** if $w_t[a] = 0$, then $\lambda_t[a] = 1,\ \eta_t[a] = 0$; the slot is unchanged.
2. **Bounded content:** $0 \le \eta_t[a] < 1$ makes each update a convex combination of old content and v_t.
3. **Adaptive retention:** larger retained mass slows new writes; larger γ_t reduces retention.

Readout and interpretation

$o_t = \sum_a r_t[a] s_t[a]$, followed by per-head RMSNorm and an output gate. Each slot update is also a gradient step on $\frac{1}{2}\|s - v_t\|_2^2$ with step size $\eta_t[a]$.

Paper, Sec. 3.4, PDF p. 6.

Access cost is decoupled from total state capacity

Quantity	Scaling per head	Meaning
Slot count	$M = d_p^U$	fixed during a sequence
State storage	$O(M(d_v + 1))$	content plus write mass
Address search	$O(UKd_p + Ud_p \log d_p)$	no M-wide scan
State touch	$O(Kd_v)$	sparse read/write

- Training/prefill: decode in parallel, group events by slot, preserve time order within each slot.
- Autoregressive decoding: update/read the selected slots.

Paper, Sec. 4, PDF p. 6. Scaling describes the recurrent core at fixed architecture hyperparameters.

Evaluation spans language modeling and three recall tests

Training	FineWeb-Edu; Llama 2 tokenizer; 4,096-token context
Scales	340M parameters / 10B tokens; 1.3B / 100B
Baselines	Transformer++, Mamba2, GLA, GSA, Gated DeltaNet
Language	WikiText-103 ppl ↓; zero-shot task average accuracy ↑
Recall	MQAR; RULER S-NIAH at 1K–16K; nine real-document recall columns ↑
Systems	340M decode throughput on one RTX 5090 ↑

Paper, Sec. 5 and Appendix A, PDF pp. 7–9, 14.

At 340M, quality rises with K while access remains small

Model	Wiki ppl ↓	Commonsense avg. ↑	Active state ↓
Transformer++	26.81	49.4	100.7M
Mamba2	26.23	50.6	12.9M
Gated DeltaNet	26.85	50.1	8.5M
RAM-Net Top-8	26.89	50.3	0.4M
RAM-Net Top-32	**25.87**	**50.9**	1.6M

- Top-8 touches about 32× fewer state elements than Mamba2; this is an access count, *not* a throughput ratio.
- The total RAM-Net state is 26.2M elements at this scale.

Paper, Table 1 and footnote, PDF p. 7. Transformer++ active state measured at length 2,048.

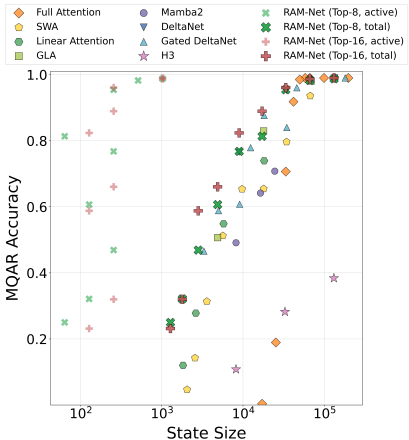
At 1.3B, RAM-Net matches the best average accuracy

Model	Wiki ppl ↓	Commonsense avg. ↑	Active state ↓
Transformer++	17.08	56.0	201.3M
Gated DeltaNet	17.09	56.4	17.0M
Mamba2	15.96	**57.8**	26.0M
RAM-Net Top-8	16.49	57.4	0.8M
RAM-Net Top-16	**15.90**	**57.8**	1.6M

- Top-16 has the lowest reported perplexity by a small margin and ties Mamba2 on the average accuracy.
- Total RAM-Net state is 52.4M elements at this scale.

Paper, Table 1 and footnote, PDF p. 7.

MQAR favors sparse addressing at a given state size



Original paper Fig. 4, PDF p. 7; Sec. 5.

- MQAR: synthetic key–value recall; models trained from scratch.
- RAM-Net Top-8/16 lies near the upper envelope by *total* state size.
- Pale/dark RAM-Net markers denote *active*/*total* state.

Interpretation: partitioned slots help limit interference between unrelated writes.

S-NIAH averages favor RAM-Net at both model scales

RULER single-needle accuracy averages 3 task types $\times$ 5 lengths (1K–16K). All models trained at 4K context.

Scale	Transformer++	Best other recurrent	RAM-Net
340M	48.6	Gated DeltaNet 51.3	**Top-32 72.9**
1.3B	56.4	Mamba2 62.4	**Top-16 70.1**

- S-NIAH-1 has repeated filler; RAM-Net stays near 100% through 16K.
- The more diverse S-NIAH-2/3 settings are substantially harder.

Paper, Table 2 and Sec. 5, PDF p. 8. Values are accuracy percentages.

The difficult 16K tail exposes an addressing trade-off

	S-NIAH-2		S-NIAH-3	
Model	8K	16K	8K	16K
340M RAM-Net Top-32	56.3	4.0	9.7	1.0
1.3B RAM-Net Top-16	49.0	18.7	18.3	6.3

Why this matters

Sparse access reduces collision, but a read misses relevant content if its query does not overlap the original write slots.

The paper links the easy S-NIAH-1 case to repeated filler sharing addresses; diverse filler makes interference and routing harder.

Paper, Table 2 and Sec. 5, PDF p. 8; limitation discussed on PDF p. 10. Values are percentages.

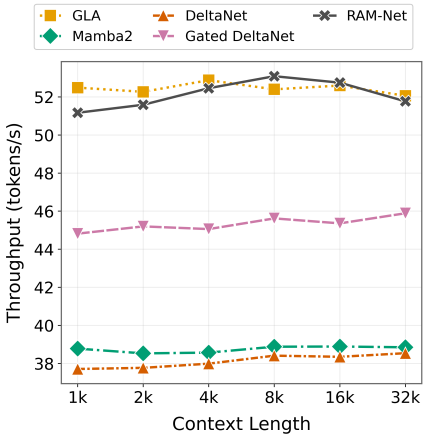
Real-document recall still trails full attention

Scale	Transformer++	Best other recurrent	RAM-Net
340M	**35.1**	Mamba2 26.9	Top-32 32.0
1.3B	**49.8**	Gated DeltaNet 39.5	Top-8 40.6

- Average over FDA, SWDE, NQ at 512/1K, plus full-document SQuADv2, TriviaQA, and DROP.
- RAM-Net leads the listed recurrent models at both scales, while per-token KV storage retains an advantage on document detail.

Paper, Table 3, PDF p. 8; Sec. 5, PDF p. 9. Values are average accuracy percentages.

Sparse access translates into competitive decoding speed



- 340M models; one NVIDIA RTX 5090.
- Contexts from 1K to 32K; output in tokens/s.
- RAM-Net Top-8 stays near GLA and above Mamba2, DeltaNet, and Gated DeltaNet across the shown range.

GLA is fast with a small *total* state; RAM-Net accesses a small part of a larger state.

Original paper Fig. 5, PDF p. 9; Sec. 5. Transformer++ is not shown in this plot.

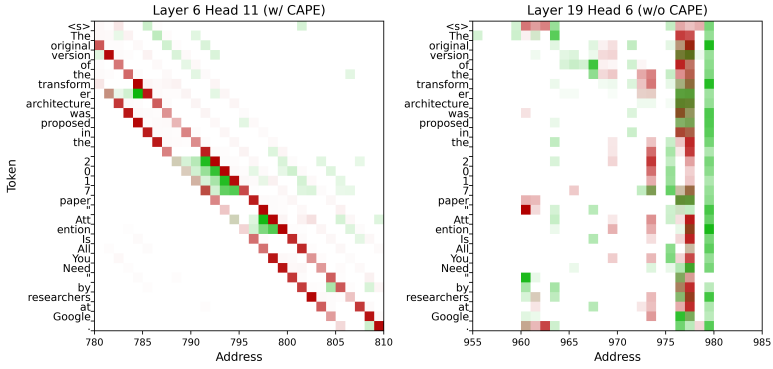
Ablations identify more than a slot-count effect

340M variant	Wiki ppl ↓	S-NIAH-2 @4K ↑	S-NIAH-3 @4K ↑
Top-8 default	26.89	83.7	18.3
Top-32	**25.87**	**94.0**	**44.7**
No output RMSNorm	28.87	16.3	12.3
NoPE only	28.04	43.0	8.7
Renormalize Top-K	26.83	63.7	12.0
Slot balancing loss	27.10	11.3	5.7

Reading: readout scaling, position-aware heads, and uneven slot usage all contribute to retrieval.

Paper, Table 5, PDF pp. 15–16. Values are percentages except perplexity.

Access traces show complementary head behavior



- CAPE head: diagonal drift from position-dependent shifts.
- NoPE head: vertical bands of reused, content-driven slots.

Original paper Fig. 7, PDF p. 16; interpretation in Appendix C, PDF p. 17.

Take-home: sparse access buys recall, with routing limits

1. A fixed state can be partitioned into many addressable slots. Product Softmax and exact Top-K select a few efficiently.
2. CAPE and GSU make position-aware access and sparse forgetting compatible with the slot array.
3. Reported retrieval gains are strong on MQAR and S-NIAH averages; language modeling remains competitive.

Main boundary in the reported evidence

Queries can miss relevant write slots. Difficult 16K retrieval remains weak in absolute terms; real-document recall trails full attention. Experiments stop at 1.3B parameters.

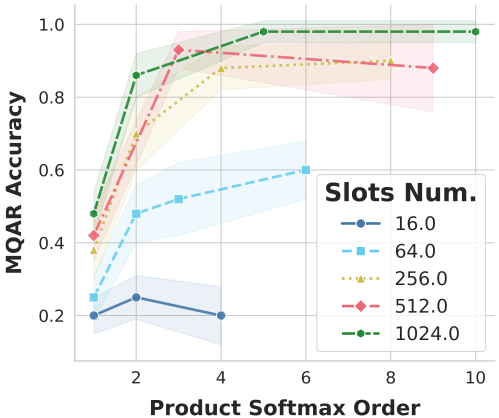
Paper, Secs. 5, 7–8, PDF pp. 7–10.

Sources

- Kaicheng Xiao, Haotian Li, Liran Dong, Guoliang Xing. *RAM-Net: Linear-Time Sequence Modeling with Sparsely Addressable State*. arXiv:2602.11958v2, 2026.
- Acceptance status: authors' arXiv v2 comment; NeurIPS 2026 official list.
- Official public code repository.

Original paper figures appear on slides 3–7, 14, 18, 20, and 23; cropped panels are identified on their slides.

Backup: product structure also helps optimization



Original paper Fig. 6 and Appendix B, PDF pp. 14–15.

- MQAR accuracy generally rises with Product Softmax order U, especially at larger slot counts.
- With $U = 1$, a single M-way Softmax gives selected entries a sparse direct learning signal.
- With $U > 1$, selected slots share digit distributions.

Backup: training and initialization details

- Both scales use FineWeb-Edu, Llama 2 tokenizer, AdamW, 4K context, cosine decay, and 0.5M-token global batches.
- 340M: 10B tokens, peak LR 10^{-3}, 500M warmup tokens.
- 1.3B: 100B tokens, peak LR 4×10^{-4}, 1B warmup tokens.
- Per-digit, per-head learnable temperature correction begins from base $\tau = 2.2$.
- Biases initialize output gate near open, accelerate early forgetting, and break address symmetry.

Paper, Appendix A, Table 4, PDF p. 14.