A Hippocampus for Linear Attention

An Exact Memory for What the Recurrent State Forgets

Paper: Wanyun Cui
Shanghai University of Finance and Economics

arXiv:2607.02303v1 | 2 July 2026
Research discussion prepared 22 September 2026

A bounded exact KV memory complements a compressed recurrent state.

A fixed recurrent state can lose specific facts

Recurrent memory

A fixed matrix $S \in \mathbb{R}^{d_k \times d_v}$ compresses the prefix.

- Bounded state as context grows.
- Shared key directions cause interference.
- Distant exact associations can be lost.

Full attention

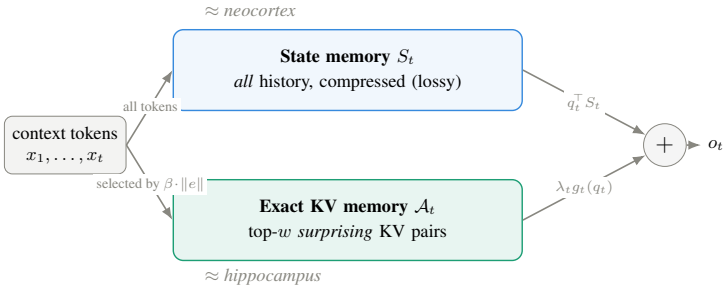
Every causal token remains available as a KV pair.

- KV storage grows with length.
- Exact extraction remains stronger on some tasks.
- HOLA adds a small selective KV store to GDN.

Research question: which facts should bypass state compression?

Paper, Secs. 1–3.1, PDF pp. 1–4. Hippocampus/neocortex are design analogies.

Two memories provide complementary readouts



$$o_t = S_t^\top q_t + \lambda_t g_t(q_t)$$

State: a linear estimator. Cache: a bounded non-parametric read.

Exact KV storage does not guarantee correct recall.

Original Fig. 2, PDF p. 4; Definition 2 / Eq. (3), p. 5. Column-vector convention in these slides.

The delta rule already computes a useful write signal

$$e_t = v_t - \alpha_t S_{t-1}^\top k_t, \qquad S_t = \alpha_t S_{t-1} + \beta_t k_t e_t^\top$$

- e_t: value prediction residual after decay α_t.
- $\beta_t \in [0, 1]$: how strongly the correction is written.
- State-path keys satisfy $\|k_t\|_2 = 1$.

Cache score: the committed residual update

$$m_t = \|\beta_t k_t e_t^\top\|_F = \beta_t \|k_t\|_2 \|e_t\|_2 = \beta_t \|e_t\|$$

This measures the correction, not the full state change including decay.

Paper, Eqs. (1), (5)–(6), PDF pp. 4–5; GDN instantiation, Sec. 3.5, p. 6.

Keep high-write tokens regardless of their age

Persistent cache

Keep the top-w causal KV pairs by their insertion-time score m_i. Default: $w = 64$ per layer.

Teaching example	$\|e\|$	β	$m = \beta\|e\|$
Token A	10	0.1	1.0
Token B	2	0.8	**1.6**

- Fixed scores allow online or blockwise top-w merging.
- Scores are not refreshed when queries or memory contents change.
- Large updates are a heuristic for future usefulness, not a guarantee.

Paper, Sec. 3.3, PDF pp. 5–6. Numerical example is explanatory, not a paper experiment.

The visible set has at most 321 entries

$$\mathcal{V}_t = \text{persistent cache} \cup \text{causal current-block tokens} \cup \text{null sink}$$

64 persistent	Up to 256 current	1 sink
Highest-score KV pairs retained across blocks.	Tokens from the current processing block, causally masked.	An additional null entry for read stability.

- $w + C + 1 = 64 + 256 + 1 = 321$ bounds the visible set.
- Only w is the persistent distant-token budget.
- The paper specifies cache semantics, not full block-level code.

Paper, Sec. 3.3, PDF p. 6; Sec. 5, p. 10; Appendix A, p. 12.

An exact cache needs a sufficiently sharp read

$$\tilde{q}_t = \mathrm{RMSNorm}_\gamma(q_t), \qquad \tilde{k}_j = \mathrm{RMSNorm}_\gamma(k_j)$$

$$o_t^{\mathrm{cache}} = \sum_{j \in \mathcal{V}_t} \mathrm{softmax}_j\left(\frac{\tilde{q}_t^\top \tilde{k}_j}{\sqrt{d}}\right) v_j$$

Cache normalization	Approximate norm	Logit scale
Unit L2	1	$\tau \cos\theta / \sqrt{d}$
RMSNorm, $\gamma \approx 1$	$\sqrt{d}$	$\sqrt{d} \cos\theta$

Learn Q/K scales; freeze $\tau = 1$. Keep the state path unit-normalized.

Paper, Eq. (7) and Sec. 3.4, PDF p. 6. Scale approximation ignores epsilon and learned anisotropy.

Read normalization is separated from update stability

The state update contains the operator $I - \beta k k^\top$.

$$\lambda_{\parallel k} = 1 - \beta \|k\|^2, \qquad \lambda_{\perp k} = 1$$

Unit-norm state key

$\lambda_{\parallel k} = 1 - \beta \in [0, 1]$.
GDN additionally applies decay α_t.

RMS-sized state key

$\lambda_{\parallel k} = 1 - \beta d$ can have large magnitude.
Example: $d = 256$, $\beta = 0.5$ gives -127.

Sharpen the cache read while preserving the state update's normalization.

Paper, Sec. 3.4, PDF p. 6. Eigenvalue calculation and numerical example are explanatory.

The strongest comparison shares the same GDN backbone

Main scale	340M; 24 layers; width 1024; 4 heads, dim 256
Training	15B SlimPajama tokens; context 2048
Recipe	AdamW; peak LR 4×10^{-4}; 8×A800
Cache	$w = 64$, $C = 256$, RMSNorm, frozen $\tau = 1$
Internal controls	GDN anchor; HOLA+recency; HOLA
Uncertainty	Main-scale results use a single seed

- Recency changes only which tokens the matched cache retains.
- Transformer++, GLA/GSA, DeltaNet/KDA and one GDN row are borrowed from recipe-matched papers.

Paper, Sec. 4.1 / Table 1, PDF p. 7; Sec. 5, p. 10; Appendix A, p. 12.

HOLA lowers perplexity in the reported 340M comparison

Model	Wiki PPL ↓	LAMBADA PPL ↓	Provenance
Transformer++	26.88	42.15	Borrowed
KDA	26.18	31.37	Borrowed
GDN anchor	27.32	30.95	Internal
HOLA+recency	25.04	32.33	Internal
HOLA	**22.92**	**30.26**	Internal

Against the internal GDN anchor

Wiki: 27.32 → 22.92, a **16.1% relative reduction**.
LAMBADA: 30.95 → 30.26, a 2.23% relative reduction.

These are perplexity changes, not accuracy percentage points.

Selected rows from Table 1, PDF p. 7. Relative changes calculated from reported values.

Retrieval improves; commonsense scores remain close

Model	FDA ↑	SWDE ↑	Six-task mean ↑
Transformer++ (borrowed)	**46.1**	25.9	42.34
GDN anchor	11.7	29.0	42.54
HOLA+recency	16.9	29.9	**43.17**
HOLA	20.1	**35.9**	42.85

- FDA: +71.8% relative to GDN, but still below full attention.
- SWDE: +23.8% relative to GDN; above this full-attention row.
- Small commonsense differences have no reported error bars.

Table 1, PDF p. 7. Six-task mean mixes accuracy and acc_norm; retrieval scorer details are not fully specified.

Matched recency isolates the value of token selection

Hold architecture and memory budget fixed

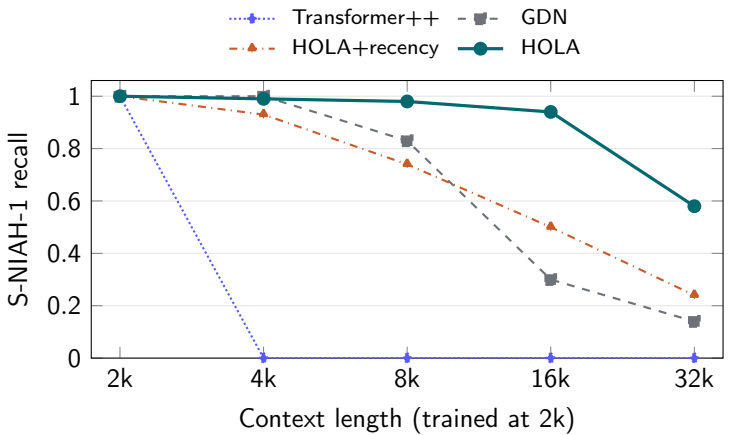
Same w, chunk, sharpened read, gate and kernel.
Change eviction: most recent tokens $\rightarrow$ largest $\beta\|e\|$.

340M result	Recency	HOLA
Wiki PPL $\downarrow$	25.04	**22.92**
S-NIAH-1 at 32k $\uparrow$	0.24	**0.58**
Commonsense six-task mean $\uparrow$	43.17	42.85

Recency already helps Wiki PPL versus GDN; selection adds further gains.

Sec. 4.5, PDF pp. 8–9; Table 1 and Fig. 1. Recency quality evaluation uses full recomputation; this is not a decode-speed comparison.

Single-needle recall remains stronger out to 32k



At 32k

HOLA: 0.58
Recency: 0.24
GDN: 0.14

+44 percentage points versus GDN.

Still far from perfect recall.

Redrawn from Fig. 1b, PDF p. 1, using exact coordinates in fig_teaser.tex. RULER: Mistral tokenizer, limit 100. This Transformer++ checkpoint is not adapted for length extrapolation.

Harder RULER tasks reveal clear limits

Task / context	GDN	Recency	HOLA
S-NIAH-2 / 2k	0.38	0.83	**1.00**
Multi-query / 2k	**0.24**	0.22	0.18
Multi-key-1 / 4k	0.17	0.13	**0.30**
S-NIAH-2 / 8k	0.09	0.04	**0.35**
S-NIAH-3 / 8k	**0.07**	0.02	0.05
Multi-query / 8k	0.02	0.01	**0.08**

- HOLA wins many cells, but not every task or length.
- Several hard tasks remain near the floor at 8k.
- Reported needle results do not cover the full RULER task suite.

Selected cells from Table 3, PDF p. 9. Accuracy, higher is better. Setup mentions VT but reports no VT result.

Perplexity gains repeat across three tested scales

Scale	Train tokens	GDN PPL	HOLA PPL	Reduction
46M	0.50B	71.0	59.5	16.20%
170M	6.22B	35.98	30.51	15.20%
340M	15.0B	27.32	22.92	16.11%

- Within each scale: matched GDN/HOLA backbone and recipe.
- 46M: FineWeb-Edu, context 4096, width 512, 12 layers.
- 170M/340M: SlimPajama, context 2048, width 1024.

This is cross-scale consistency, not a parameter-only scaling law.

Table 2, PDF p. 8; Appendix A / Table 6, p. 12. Relative reductions calculated from displayed values.

At 46M, committed residual beats residual alone

Flat cache read; same backbone and cache budget for cached variants.

Eviction score	Far ↑	4 keys ↑	Wiki PPL ↓
GDN, no cache	0.55	0.41	70.21
Cumulative attention	0.43	0.55	71.45
$\|e\|$	0.22	0.48	70.50
$\beta\|v\|$	0.42	0.54	70.76
$\beta\|e\|$	**0.67**	**0.56**	**70.10**

- Residual magnitude alone is a poor far-passkey signal here.
- The PPL gain remains small until the read is changed.

Table 4, PDF p. 9. Far: teacher-forced next-token accuracy, passkey at depth 0.1 in 4k. This is not the 340M RULER protocol.

Read normalization is the largest change in this ablation

Fixed eviction: $\beta\|e\|$. 46M component study.

Model / cache read	PPL ↓	Far	4 keys	8 keys	16 keys
GDN, no cache	70.21	0.55	0.41	0.35	0.31
Unit L2	70.10	0.67	0.56	0.41	0.31
RMSNorm-γ	**59.5**	**0.75**	**0.77**	**0.67**	**0.41**

- Wiki PPL falls by 10.60 when changing the cache read.
- Multi-key recall improves, while far-passkey accuracy is preserved.
- RMSNorm row: three-seed mean; variance is not reported.

Table 5, PDF p. 10. Recall columns: higher is better. Other rows reuse flat-read anchors. No fixed-high-temperature row is reported.

Memory overhead is bounded; speed gains are unmeasured

Additional model parameters

$L(2d + 2H) = 12{,}480$ trainable scalars.

Less than 0.004% at 340M.

Plus 96 frozen temperature scalars.

Additional decoding state

Up to $w + C = 320$ KV pairs per layer.

About 31 MB in bf16 at the stated configuration.

Peak GPU allocation, batch 1	GDN	HOLA
At 32k and at 128k, weights included	0.72 GB	0.75 GB

128k tests memory usage only. No throughput or latency table is reported.

Paper, Sec. 3.5, PDF p. 6. Reported rounded peaks imply about 4.2% extra; paper says about 5%.

The evidence supports a selective complement to GDN

- **Supported:** lower PPL and better reported retrieval at the tested scales, with a strong matched-recency control.
- **Not established:** reliable commonsense gains, optimal eviction, or superiority to a matched learned eviction module.
- **Remaining limits:** a small persistent cache, single-seed main results, and weak hard multi-needle recall.
- **Deployment questions:** full causal implementation details, larger models, and measured speed/quality trade-offs.

Paper, Secs. 3–5, PDF pp. 4–10. Evidence synthesis and deployment questions are this reading's analysis.

Design the cache's write and read together

1. **Preserve a small set of exact associations.**
   A fixed recurrent state alone loses some details.

2. **Use the update to guide retention.**
   $\beta\|e\|$ is available without an extra eviction network.

3. **Make retained associations retrievable.**
   A separate RMSNorm path makes the cache read more selective.

Most defensible claim: bounded exact memory helps, and both selection and read normalization matter.

Synthesis of Eqs. (3), (6), (7), Table 1, and Tables 4–5. No claim of perfect or universally optimal recall.

Reference and reading scope

Wanyun Cui.

A Hippocampus for Linear Attention: An Exact Memory for What the Recurrent State Forgets.

arXiv:2607.02303v1, 2 July 2026.

`https://arxiv.org/abs/2607.02303v1`

- Read all substantive sections and Appendix A from author TeX.
- Cross-checked the 12-page versioned PDF and figure coordinates.
- Original Figure 2 is cropped; Figure 1b is redrawn from its data.
- These are independent discussion slides; experiments were not reproduced.

Prepared with the arxiv-paper-reader workflow. Chinese notes, editable TeX, figures, and local build helper accompany this PDF.

Backup: training recipes differ across scales

Scale	Width	Layers	Corpus	Tokens	Context
46M	512	12	FineWeb-Edu	0.5B	4096
170M	1024	12	SlimPajama	6.22B	2048
340M	1024	24	SlimPajama	15.0B	2048

- 170M/340M: 4 heads, head dimension 256, tied embeddings, vocabulary 32000, Mistral tokenizer.
- 340M: AdamW, weight decay 0.01, cosine schedule, warmup 1000, clip 1.0, batch 0.5M tokens, one epoch.
- HOLA: $w = 64$, $C = 256$, gate init -4, RMSNorm, frozen $\tau = 1$, SDPA cache kernel.

Paper, Sec. 4.1, PDF p. 7; Appendix A / Table 6, p. 12.

Backup: full RULER snapshot at 2k and 4k

Context	Model	S1	S2	S3	MK1	MV	MQ
2k	Transformer++	1.00	1.00	0.79	0.71	0.34	0.33
	GDN	1.00	0.38	0.26	0.17	0.17	0.24
	Recency	1.00	0.83	0.89	0.26	0.21	0.22
	HOLA	1.00	1.00	0.96	0.25	0.28	0.18
4k	Transformer++	0.00	0.00	0.00	0.00	0.00	0.00
	GDN	1.00	0.52	0.23	0.17	0.14	0.19
	Recency	0.93	0.27	0.38	0.13	0.17	0.17
	HOLA	0.99	0.89	0.43	0.30	0.28	0.26

At 2k S3, HOLA (0.96) exceeds Transformer++ (0.79).

Paper, Table 3, PDF p. 9. Accuracy, higher is better. S1/S2/S3: single needle; MK1/MV/MQ: multi-key/value/query.

Backup: numerical details need careful qualification

- **Logit examples use unexplained dimensions.**
  $6.6/\sqrt{d} \approx 0.83$ suggests $d \approx 64$, while $\sqrt{d} \approx 11$ suggests another dimension. The main model uses $d = 256$.

- **The 46M GDN anchors differ.**
  Table 2 reports 71.0; Tables 4–5 report 70.21. The paper does not explain the difference.

- **A tiny gain has a rounding discrepancy.**
  BoolQ $0.548 \rightarrow 0.584$ is $+3.6$ points from displayed values; the prose says $+3.5$.

- **Mechanism claims exceed the shown ablation.**
  Table 5 has no fixed-high-temperature row or attention-entropy diagnostic.

Paper, Sec. 3.4, p. 6; Sec. 4.2 / Table 2, p. 8; Tables 4–5, pp. 9–10. These are reading/audit observations.

Backup: a conceptual causal implementation

1. Compute Q/K/V, decay α_t, and write gate β_t.
2. Predict with the decayed state; form e_t and update S_t.
3. Fix $m_t = \beta_t \|e_t\|$; maintain bounded top-w KV.
4. Read the cache plus the causal current-block prefix and sink.
5. Combine the state read and gated cache read.

Details requiring implementation-level verification

Block update timing, deduplication, score ties, per-head selection, top-w gradients, and the exact cache-gate parameterization.

Explanatory procedure based on Sec. 3, PDF pp. 4–6. The downloaded source is paper TeX, not an executable model implementation.