Key-Value Means

Transformers with Expandable Block-Recurrent Compressed Memory

A paper discussion

Paper authors: Daniel Goldstein, Navneel Singhal, Eugene Cheah

arXiv:2605.09877v5 (3 August 2026)
Reading: 22 September 2026

Exact recent tokens + compressed, optionally growing memory.

Memory capacity should be a design choice

Full attention

Keep every historical KV pair.

- State grows as $O(N)$.
- Each new query scans growing history.

Fixed-state recurrence

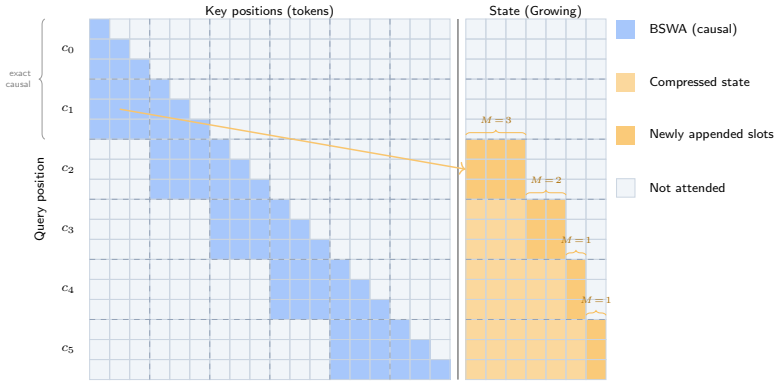
Compress everything into a fixed state.

- Constant state and amortized work.
- Novel information competes for capacity.

KVM asks: can state grow more slowly than the context?

Sec. 1, PDF pp. 1–2; Table 1. N = sequence length.

One softmax reads recent tokens and old memory



Blue: causal local window. Orange: compressed historical state.

Original Fig. 2, PDF p. 4. Illustration: C = 3, L = 6; experiments: C = 256, L = 512.

Block boundaries separate reading from compression

1. Form Q, K and V using the GPTAlpha-2 backbone.
2. Read the current state and causal BSWA window.
3. At the boundary, remove the oldest window block.
4. Append selected tokens; merge the rest into memory.
5. Reuse the updated state for the next query block.

Initialization matters

The first chunk creates C state rows. Do not merge it a second time.
The first $S = 1$ row is a protected, readable sink.

Sec. 3.2, PDF pp. 4–6; Appendix B, pp. 17–18 (equivalent shifted update order).

Readout uses one competition over both regions

$$K^A = \begin{bmatrix} \tau_{\text{state}} \hat{s}^K \\ \tau_{\text{bswa}} k_{b:e} \end{bmatrix}, \qquad V^A = \begin{bmatrix} \hat{s}^V \\ v_{b:e} \end{bmatrix}$$

$$y_u = \text{softmax}\left(\frac{q_u (K^A)^\top}{\sqrt{d_h}} + M_u \right) V^A$$

- $\hat{s}^K, \hat{s}^V$: normalized memory readouts.
- τ: learned per-head inverse temperatures for each region.
- M_u: all old state visible; future local tokens masked.

Sec. 3.2 Readout, PDF pp. 5–6 (unnumbered equations). $b = e - L$; d_h = head dimension.

Append the tokens least represented by old memory

$$s_j = \max_i \bar{k}_j \hat{s}_i^{K\top}, \qquad A_e = \text{indices of the } n_{\text{append}} \text{ smallest } s_j$$

- A preset budget determines *how many* rows to add.
- Similarity determines *which* overflow tokens get new rows.
- Copy $\bar{k}_j$, v_j, and the initial value radius $\|v_j\|_2$.
- Then merge the remaining tokens into the expanded state.

Content selection is adaptive; capacity scheduling is preset

No learned, content-dependent growth budget is evaluated here.

Sec. 3.2 Append, PDF p. 6; Appendix G, p. 22. Overflow token index: j.

Merge uses hard routing and a learned write weight

$$g_j = 1 + \mathrm{ELU}(x_j W_g), \qquad \pi(j) = \arg\max_{i \geq 1} \bar{k}_j \, \mathrm{LN}_s(s^K_{+,i})^\top$$

$$s^K_{\mathrm{new},i} = s^K_{+,i} + \sum_{j:\pi(j)=i} g_j \bar{k}_j, \quad s^V_{\mathrm{new},i} = s^V_{+,i} + \sum_{j:\pi(j)=i} g_j v_j$$

- Route into old rows or rows appended in this same step.
- Exclude the protected sink from merge targets.
- Positive g_j controls update strength, not the target ranking.
- Batch assignments precede accumulation; old state is not decayed.

Sec. 3.2, PDF pp. 5–6. s_+ is the post-append state; W_g is d by 1 for each head.

Value radius preservation protects readout scale

$$\hat{s}_i^K = \mathrm{LN}_s(s_i^K), \qquad \hat{s}_i^V = \rho_i \frac{s_i^V}{\max(\|s_i^V\|_2, \epsilon_{\mathrm{norm}})}$$

- Persist weighted sums; normalize just before reading.
- ρ_i is set when a slot is created and never updated.
- No per-slot token count is needed for this normalized readout.

Reader's teaching example

Averaging $(1, 0)$ and $(0, 1)$ shrinks the norm to $1/\sqrt{2}$.
Restoring norm preserves scale, but cannot recover both input vectors.

Sec. 3.2, PDF pp. 5–7; Appendix G, p. 22. Normalization is not lossless compression.

Local position survives; old rotary content is removed

$$\bar{k}_t = \mathrm{LN}_s\left(k_t \,\mathrm{diag}(\underbrace{0,\ldots,0}_{r},\underbrace{1,\ldots,1}_{d_h-r})\right)$$

- Queries and local keys use partial RoPE (usually half the channels).
- Overflow keys lose the original rotary-subspace content.
- A single query/projection family can read both regions.
- This sacrifices some key expressivity; it is not inverse rotation.

Formula-reading caveat: affine LayerNorm follows zeroing; final channels need not remain zero. Strict position invariance is not proved.

Eq. (1), PDF p. 5; Appendix G, pp. 22–23. Caveat is reader analysis.

State growth sets the asymptotic cost

Strategy	State	Prefill	Decode/token
KVM fixed	$O(1)$	$O(N)$	$O(1)$
KVM sqrt	$O(\sqrt{N})$	$O(N^{1.5})$	$O(\sqrt{N})$
Full attention	$O(N)$	$O(N^2)$	$O(N)$

- Fixed chunk/window sizes and head dimensions; amortized decode.
- Reader derivation: $m(t) \propto t^{\alpha}$ gives $O(N^{1+\alpha})$ prefill.
- Within-chunk parallelism does not remove state dependencies.
- Runtime state size is not total training peak memory.

Table 1, PDF p. 2; Appendix B, p. 18. General exponent and memory caveat: reader analysis.

The novelty lies in a combination of choices

- **Versus OVQ:** joint local/state softmax, growable rows, gated sums and JIT normalization without counts.
- **Versus fixed-state recurrence:** capacity can grow with context.
- **Versus post-hoc KV merging:** train the model to use compressed memory; do not preserve full-attention outputs exactly.
- **Implementation:** standard operations are sufficient; reported speed results use custom Triton kernels.

Sec. 2, PDF pp. 2–3; Appendix A, pp. 16–17; Sec. 5, pp. 8–9.

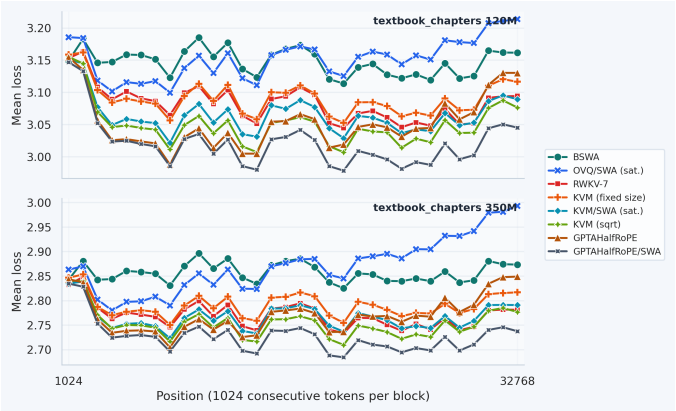
Evaluation covers small trained models and length extrapolation

Model / tokens	120M / 3B; 350M / 7.8B
Data / training context	Prolong / 8K
KVM window	$C = 256$; two chunks ($L = 512$)
Memory schedules	Fixed 256; $16\sqrt{N}$; saturating hybrid

- GPTA-2 full attention, BSWA, RWKV-7 and OVQ/SWA baselines.
- TextbookChapters loss and NIAH evaluated out to 32K.
- Full RULER at **4K**; LongBench **few-shot subset**.
- RWKV-7 uses its own backbone; comparisons are not all module-only.

Sec. 4, PDF p. 7; Appendix D, p. 19. Hybrid state caps approach 1024; alternate SWA = 256.

Stable long-position loss does not guarantee retrieval



Top: 120M. Bottom: 350M. Lower loss is better; each point averages 1024 tokens. KVM remains competitive beyond 8K; GPTA HalfRoPE/SWA remains strong.

Original Fig. 4, PDF p. 8. No exact point values or confidence intervals are reported.

Growing state rescues difficult retrieval at 16K

350M model	S1 32K	S2 16K	S3 16K	S3 32K
RWKV-7	12.6	2.0	1.2	0.4
GPTA-2	22.4	45.2	37.6	5.8
KVM-256	98.8	3.6	3.0	2.4
KVM-sqrt	98.4	71.0	**77.6**	**36.6**
OVQ/SWA	45.2	17.6	8.6	2.6

- S3 16K: 3.0 $\rightarrow$ 77.6, a **74.6 percentage-point** increase.
- S2/S3 use essay distractors; novelty is the authors' explanation.
- Even growing memory degrades at 32K. Retrieval is not solved.

Selected entries, Table 2, PDF p. 8. All scores higher is better; trained context is 8K.

Retrieval gains depend on task and model scale

120M KVM-sqrt, 32K

S1: 99.6
S2: 4.2
S3: 2.6

Growing state alone is insufficient for strong difficult retrieval.

350M, S2 at 32K

KVM-sqrt: 17.0
GPTA-2: 20.8
GPTA HalfRoPE/SWA: 23.6

The same ranking does not hold across tasks and lengths.

Scale comparison changes both parameter count and training tokens.

Table 2, PDF p. 8; training setup, p. 7. Scores higher is better.

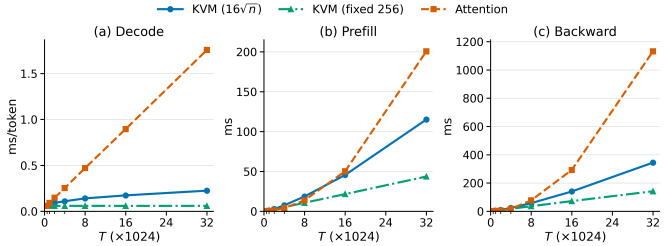
Competitive averages still leave a RULER gap

350M model	Short avg.	LB subset	RULER-4K
GPTA-2	47.2	25.1	47.2
KVM-256	47.3	23.7	33.2
KVM-sqrt	47.7	25.0	38.6
KVM/SWA	47.3	25.2	40.7
OVQ/SWA	46.9	21.0	29.0

- KVM-sqrt closely matches GPTA-2 on this LB subset average.
- It trails GPTA-2 by **8.6 points** on full RULER at 4K.
- Small short-task differences lack reported multi-seed uncertainty.

Table 2, PDF p. 8; Table 4, p. 20. All columns higher is better; short avg. excludes perplexity.

Speed improves at long contexts within a defined kernel scope



32K speedup	Decode	Prefill	Backward
KVM-256	29.77×	4.59×	7.99×
KVM-sqrt	7.88×	1.76×	3.30×

MI325X; one layer; 32 heads × 128 dims; batch 8; BF16 I/O.
Includes routing/state updates; excludes projections, position encoding and other layer work.

Original Fig. 5 and Sec. 5, PDF pp. 8–9. Baseline: PyTorch Flash SDPA / ROCm AOTriton.

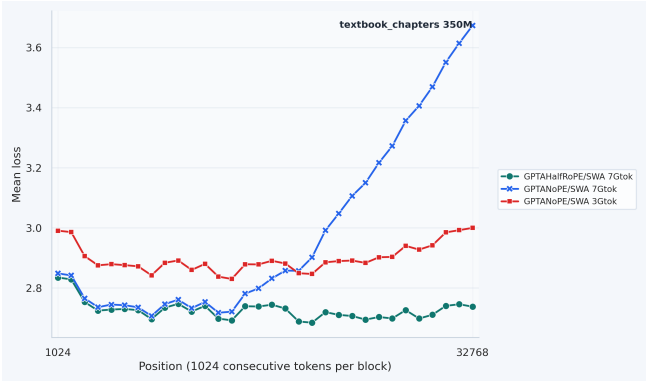
Value normalization matters far more for long retrieval

120M KVM-256	S1 32K	RULER-4K	Short avg.
Baseline	98.4	25.2	42.0
No sink	79.6	19.1	41.9
No head temperatures	98.8	25.5	42.2
No value-length norm	**3.6**	13.4	42.6
No merge gate	87.8	20.3	42.5

- Removing value normalization costs 94.8 points on S1 32K.
- Short benchmarks largely miss this failure.
- Temperature gains are mixed; key JIT norm is not separately ablated.

Table 3, PDF p. 9; Table 5, p. 20. Selected reported entries; all columns higher is better.

Length extrapolation has several distinct meanings



- HalfRoPE stabilizes position-wise loss.
- NoPE can still win on specific retrieval tasks.
- More NoPE training improves some scores but worsens long-position loss.

These tests use GPTA-2/SWA, not an isolated KVM position ablation.

Original Fig. 6, p. 21; Tables 6–7, pp. 20–21. Figure says 7Gtok; tables specify 7.8Gtok.

The evidence supports a trade-off, with open questions

- **Scale:** 120M/350M language-model experiments, not large deployed models.
- **Compression:** no lossless guarantee, error bound or capacity theorem.
- **Systems:** no end-to-end throughput or full peak-memory study.
- **Schedules:** fixed during training; adaptive capacity and train/test changes remain future work.
- **Transfer:** GQA/MLA combinations and distillation are proposed, not demonstrated.

Sec. 7, PDF pp. 9–10; evidence boundaries from Secs. 4–6 and Appendices D–G.

Take home: grow memory when fixed capacity becomes limiting

1. **Mechanism:** one softmax over exact recent KV and compressed old KV; append novel content, merge similar content.
2. **Result:** growing state substantially improves difficult retrieval in some settings, while reducing long-context mixing cost.
3. **Boundary:** gains depend on task, scale and budget; full attention still wins some important comparisons.

Discussion

Which workload is limited by memory capacity, and which by local modeling or compute overhead?

Synthesis of Secs. 3–7. Suggested main presentation: about 25 minutes; backup follows references.

Paper and reproducibility entry points

Daniel Goldstein, Navneel Singhal, Eugene Cheah.

Key-Value Means: Transformers with Expandable Block-Recurrent Compressed Memory.

arXiv:2605.09877v5, revised 3 August 2026.
arxiv.org/abs/2605.09877v5

Code linked by the paper: featherless-ai/KVM-paper
Models linked by the paper: featherless-ai/kvm-paper

This discussion reads the version-matched PDF and author TeX, including Appendices A–G. Original figures are reused; selected tables are re-typeset. Experiments were not rerun.

Backup: budget growth is monotone and supply-limited

$$M^\star(e) = \max\{m, \min(\mathcal{B}(e), m + C)\}, \quad n_{\text{append}} = \min(M^\star(e) - m, |\Omega_e|)$$

- Never delete state rows; add at most one overflow chunk.
- Initialize with C rows before applying growth updates.

Choice	Budget behavior	Evaluated configuration
Fixed	256 slots	All KVM layers
Power law	$16\sqrt{N}$	All KVM layers
Saturating	Approaches 1024	KVM/SWA hybrid

Sec. 3.2, p. 6; schedules summarized from Fig. 3, p. 5 and Sec. 4, p. 7. m = current rows.

Backup: row counts are not measured memory bytes

Reader calculation at $N = 32768$, with $L \leq 512$

KVM-sqrt: $16\sqrt{32768} \approx 2896$ state rows.
State plus window: ≈ 3408 rows, or 10.4% of full KV rows.

KVM-256: at most $256 + 512 = 768$ rows, or 2.34%.

- Excludes radius scalars, dtype differences and temporary tensors.
- Training may retain intermediate states for backpropagation.
- Budget rounding and block timing affect exact counts.

Reader calculation from the schedules in Sec. 4, p. 7; not an empirical GPU-memory measurement.

Backup: backbone and training are part of the comparison

- GPTAlpha-2: Q/K LayerNorm with bias; token shift; value residual from the first layer; partial RoPE; squared-ReLU channel mixer.
- CompleteP ($\alpha = 1$); AdamC with $\beta_1 = .9$, $\beta_2 = .95$, $\epsilon = 10^{-8}$.
- Base LR 2×10^{-3} and weight decay .2, tuned at 120M/3B; longer-run scaling by $1/\sqrt{N_{\text{steps}}}$.
- Batch: 524,288 tokens. Warmup: 200 steps, then linear decay to zero. RoPE base: 10,000.
- No weight decay on scalar/vector parameters; no reported quality multi-seed intervals.

Appendices C–D, PDF pp. 18–19. The speed-benchmark head count is not the training-model specification.

Backup: reported speed crossovers differ by operation

First faster tested length	Decode	Prefill	Backward
KVM-256	1K	8K	4K
KVM-sqrt	1K	16K	8K

- Routing and state updates create overhead at shorter contexts.
- Figure bands span min/max run medians, not confidence intervals.
- The largest ratio belongs to fixed-state decode, not all operations.
- Hardware, batch size and kernel implementation remain part of the result.

Sec. 5 and Fig. 5, PDF pp. 8–9; single-layer sequence mixing on AMD MI325X.