Muir: Linear Attention with Sensitivity-Allocated Gating and Matched-Filter Readout

A paper discussion on fixed-state memory, readout, and TPU training

Tushaar Gangavarapu Gil Tabak Clemens JS Schaefer

arXiv:2611.08376v1 | 8 November 2026

Publication: arXiv (2026); no formal venue found. Google Research record
Code: JAX reference in Appendix J; no standalone repository identified. Paper

Paper identity: arXiv v1; publication status checked 2026-11-08.

Muir treats three bottlenecks as one design problem

Gate capacity	State readout	Chunkwise training
GDN-2 uses dense erase and write projections, even though state updates respond much more to writes.	Slow-decaying rows accumulate more energy and can dominate unweighted queries.	Large chunks suit TPU matrix units, but decay division becomes unstable and triangular solves become sequential.
Allocate rank by sensitivity.	Reweight rows when reading.	Pivot and merge blocks.

Core claim: better use of a fixed-size state comes from how it is written, read, and computed.

Paper, Sec. 1, PDF pp. 1–3; Fig. 1, p. 2.

The gated Delta Rule stores a key-value association in a matrix state

For one head, $S_t \in \mathbb{R}^{d_k \times d_v}$ and normalized $k_t, q_t \in \mathbb{R}^{1 \times d_k}$:

$$S_t = \left(I - k_t^\top (b_t \odot k_t)\right) \mathrm{Diag}(\alpha_t) S_{t-1} + k_t^\top (w_t \odot v_t), \qquad o_t = q_t S_t.$$

- α_t: channel-wise decay of the previous state rows.
- b_t: erase the current key's prediction from the decayed state.
- w_t: scale the incoming value write; g_t gates the normalized output.
- State size is fixed with sequence length; key overlap creates interference.

Paper, Sec. 2.1, Eq. (1), PDF p. 3.

Erase perturbations are filtered by the current key and stored row energy

$$D_b S_t[\delta b] = -k_t^\top \left((\delta b \odot k_t)\hat{S}_t\right), \qquad D_w S_t[\delta w] = k_t^\top (\delta w \odot v_t).$$

Under isotropic unit-variance perturbations:

$$\mathbb{E}\|D_b S_t\|_F^2 = \sum_j k_{t,j}^2 \|\hat{S}_{t,j,:}\|_2^2, \qquad \mathbb{E}\|D_w S_t\|_F^2 = \|v_t\|_2^2.$$

A write changes the state whenever a value arrives. Erasure matters when the key activates a row that already holds energy.

Trained keys use about 6 of 128 channels effectively. Measured state sensitivity is 13.3× larger for writes than erasures.

Paper, Proposition 3.1 and Remarks 3.2–3.3, PDF pp. 5–6; trained diagnostics, Fig. 10 / Table 3, pp. 22–23.

Channel decay creates unequal state-row energies

For key energy $c_j = \mathbb{E}[k_{t,j}^2]$, squared decay $\lambda_j = \mathbb{E}[\alpha_{t,j}^2]$, mean erase $\bar{b}_j$, and write energy σ_v^2:

$$m_j^* \approx \frac{c_j \sigma_v^2}{1 - \lambda_j + 2\lambda_j \bar{b}_j c_j} = c_j \sigma_v^2 \, \tau_{\mathrm{eff},j}.$$

- m_j^* is the expected squared norm of state row j.
- $\tau_{\mathrm{eff},j}$ is its effective memory horizon.
- Slow rows collect more writes; fast rows retain mostly recent inputs.

Evidence

At 166M, predicted and measured row energies have median within-head Spearman $r_s = 0.960$ in Muir. At 1.34B, raw row-energy p90/p10 spread is 17.2×.

Paper, Proposition 3.4, PDF p. 6; mechanistic validation, Fig. 11 / Table 3, p. 23.

The matched filter balances surviving signal against row interference

If a key-value pair is queried after lag ℓ, the paper models MSE as

$$\mathcal{L}_\ell(\gamma) = \sigma_v^2 \left(1 - \sum_j \gamma_j c_j \bar{a}_{\ell,j}\right)^2 + \sum_j \gamma_j^2 c_j m_j^*.$$

The optimal direction is

$$\gamma_j^*(\ell) \propto \frac{\bar{a}_{\ell,j}}{m_j^*} \approx \frac{\lambda_j^{\ell/2}}{m_j^*}.$$

- The numerator keeps channels whose target signal survives to the query.
- The denominator suppresses channels with high interference energy.
- Different lags imply different optimal filters; static and dynamic readout serve different query mixes.

Paper, Proposition 3.6 and Remark 3.7, PDF pp. 7–8.

Static readout is a horizon average, not full whitening

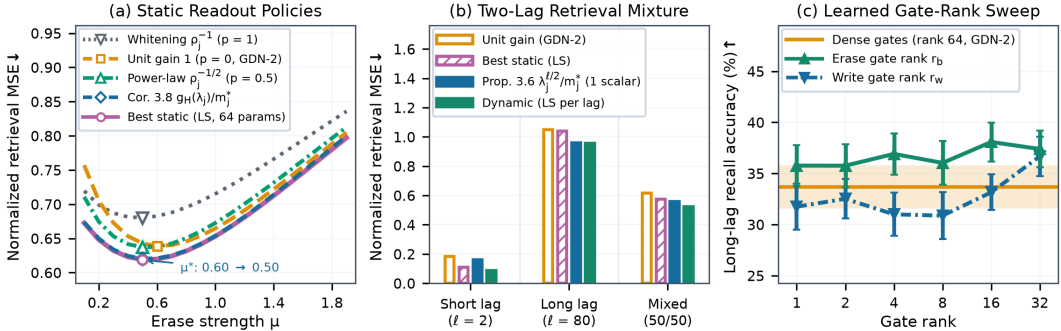
For geometric retrieval lags with mean horizon H, the static gain is

$$\gamma_j^{\mathrm{stat}} \propto \frac{g_H(\lambda_j)}{m_j^*}, \qquad g_H(\lambda_j) = \frac{\sqrt{\lambda_j}}{1 + (H-1)(1-\sqrt{\lambda_j})}.$$

- Dividing by row RMS alone over-amplifies fast channels whose target signal has already decayed.
- The trained static gain follows an intermediate power law $\gamma_j \propto \rho_j^{-p}$ with median $p \approx 0.35$.
- Readout suppresses interference at query time, allowing lower write-time erasure and longer retention.

Paper, Corollary 3.8 and Proposition 3.10, PDF pp. 8, 43; trained exponent, Fig. 11, p. 23.

Controlled memory experiments support the theory

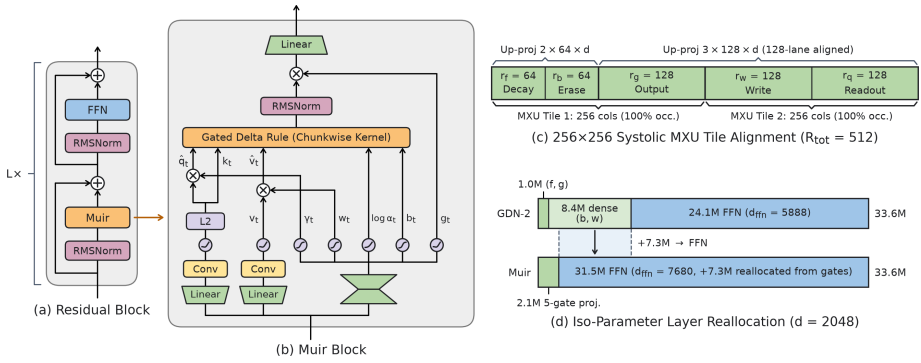


- At mean lag $H = 27$, the theory-based filter matches the static least-squares oracle and shifts optimal erase strength from 0.60 to 0.50.
- Low-rank erase gates improve long-lag recall; overly compressing the write gate hurts.

Paper, Sec. 3.2 / Fig. 2, PDF pp. 8–9; synthetic summaries over 20 seeds.

Muir adds a readout gate and shares one projection across five gates

$$o_t = (\gamma_t \odot q_t) S_t, \qquad \gamma_t = \gamma^{\text{stat}} \odot 2\sigma(z_t^q).$$



Paper, Sec. 4.1 / Fig. 3, PDF p. 10; dynamic gain initializes to one.

Rank follows sensitivity while the total fills two TPU tiles

Gate	Decay f	Erase b	Output g	Write w	Readout q
Rank	64	64	128	128	128

- Shared down-projection has total rank $R_{\text{tot}} = 512$; gate-specific up-projections retain channel biases.
- At $d = 2{,}048$, gate projection shrinks from 9.44M to 2.10M parameters (4.5× fewer).
- Same $12d^2$ layer budget: FFN width grows from 5,888 to 7,680 (+30%).
- Without dynamic readout, its 128 ranks move to the write gate: $r_w = 256$.

Paper, Sec. 4.1 / Fig. 3, PDF pp. 10–11; full scales, Appendix G / Table 11, p. 49.

Chunkwise Delta Rule becomes a triangular solve

With relative channel decay $\pi_{t,s} = \exp(\Lambda_t - \Lambda_s)$, one chunk forms

$$T = I + L_{\text{strict}}, \qquad TZ = \begin{bmatrix} \hat{V} & \hat{K}_{\odot\pi} \end{bmatrix}, \qquad Z = \begin{bmatrix} U & Y \end{bmatrix}.$$

The chunk output and outgoing state are affine maps of its initial state S_0.

- Divide by cumulative decay to build matrix products: fast-channel prefixes underflow.
- Form every pairwise decay explicitly: materializes a $C \times C \times d_k$ tensor.
- Use forward substitution: serial work on the TPU's slower vector unit.

Paper, Sec. 2.2 and Sec. 4.2, PDF pp. 4, 11–12.

Boundary pivots keep every decay factor in range

Split any cross-block decay at the boundary pivot p:

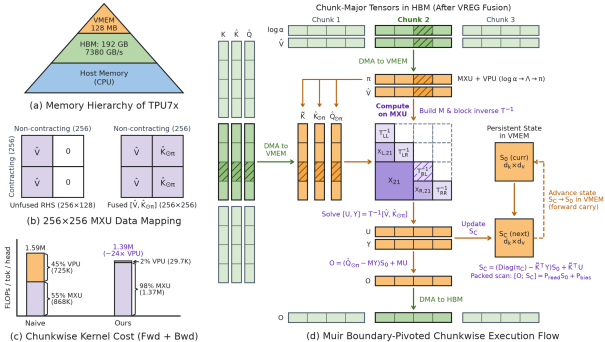
$$\pi_{t,s} = \exp(\Lambda_t - \Lambda_p) \odot \exp(\Lambda_p - \Lambda_s), \qquad s \le p < t.$$

Both log differences are non-positive, so each exponential lies in $(0, 1]$.

- Invert 8-token leaf blocks with a finite Neumann series.
- Merge adjacent triangular blocks in a five-level divide-and-conquer tree.
- Recompute pivot scalings in the custom backward pass instead of storing tree activations.

Paper, Sec. 4.2, Eqs. (7)–(9), PDF pp. 11–13; $C = 256$, leaf size 8.

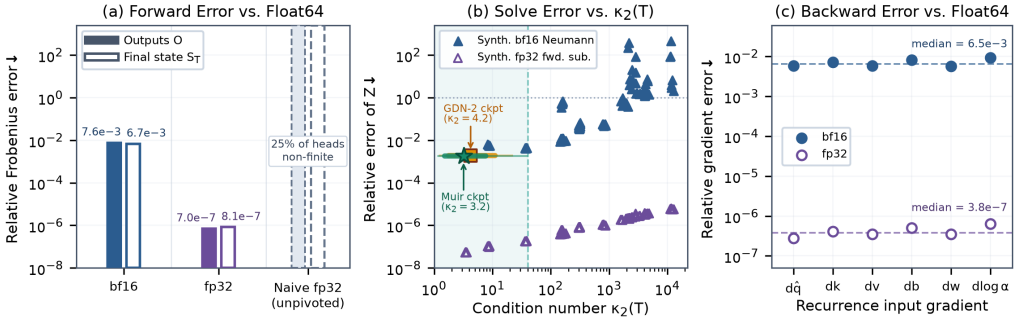
The kernel routes most work to TPU matrix units



- At $C = 256$, 98% of chunkwise FLOPs land on MXU; VPU compute falls 24×.
- Fused 256×256 solves use the full systolic tile; the state scan stays in VMEM.

Paper, Sec. 4.2 / Fig. 4, PDF pp. 13–14; Appendix F / Table 8, p. 46.

Numerical accuracy depends on the triangular system being well conditioned



- Naive unpivoted fp32 produces NaNs on 25% of tested heads; pivoted bf16 output/state errors are about 7×10^{-3}.
- Sampled training systems have $\kappa_2(T) < 40$; synthetic adversarial systems with $\kappa_2(T) \approx 12{,}865$ exceed the bf16 solver's stable range.

Paper, Sec. 5.2.4 / Fig. 9, PDF p. 20; Appendix F / Table 10, p. 47.

Pretraining compares equal-budget models under a shared recipe

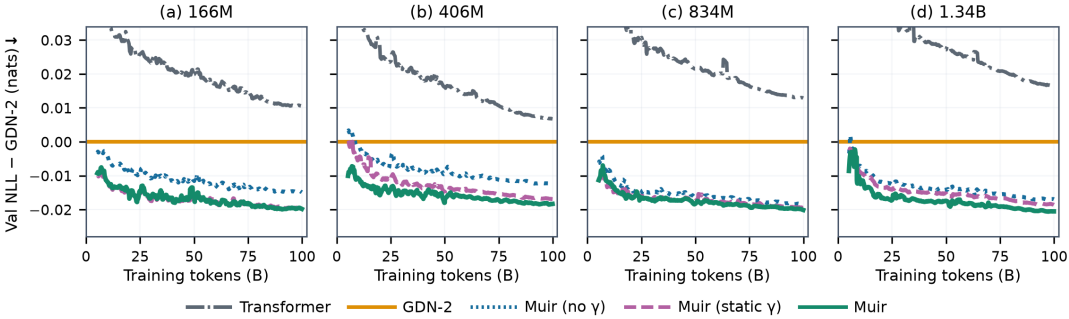
- Four parameter scales: 166M, 406M, 834M, and 1.34B.
- 100B FineWeb-Edu tokens, 4K training length, 32K tokenizer, same token order.
- 8 TPU7x chips, 524,288 tokens per step, bf16 mixed precision and fp32 state.
- Per-layer non-embedding budget: $12d^2$; recurrent state width: $d_k = d_v = 128$.
- AdamW with 1B-token warmup and cosine decay; pretraining uses fixed seed 0.

Comparisons

Transformer + Pallas Splash Attention; GDN-2; Muir without, with static, or with dynamic readout gain.

Paper, Sec. 5.1 and Appendix G, PDF pp. 14–15, 49–50.

Muir lowers validation NLL across all four scales



- Full Muir is 18.0–20.4 mnat below GDN-2 after 100B tokens.
- At 1.34B: 1.9973 vs. 2.0177 for GDN-2 and 2.0335 for Transformer.
- Muir reaches GDN-2's final validation loss by about 85B tokens.

Paper, Sec. 5.2.1 / Fig. 5 and Table 1, PDF pp. 15–16.

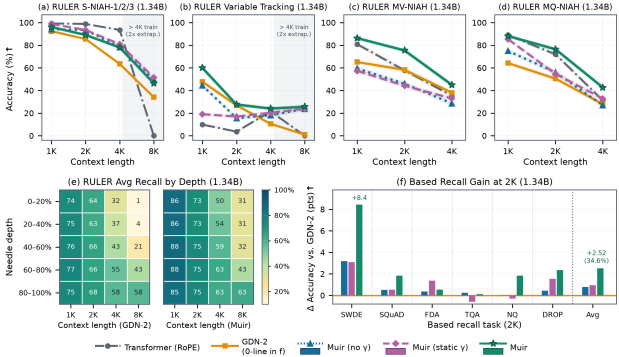
Readout helps context-sensitive tasks beyond FFN expansion

1.34B metric	Transformer	GDN-2	Muir
8-task zero-shot accuracy	52.05%	52.46%	53.84%
LAMBADA word PPL	10.30	10.38	10.00
5-shot TriviaQA accuracy	14.56%	16.48%	19.32%
Based/JRT 2K-cut average	45.92%	32.08%	34.60%

- FFN expansion explains much of the short-prompt commonsense gain.
- Dynamic readout contributes more on tasks mixing local context and distant demonstrations.
- Muir improves real-document extraction over GDN-2, while Transformer keeps the stronger overall Based/JRT average.

Paper, Table 1, PDF p. 16; Based/JRT protocol and full table, Appendix H / Table 14, p. 53.

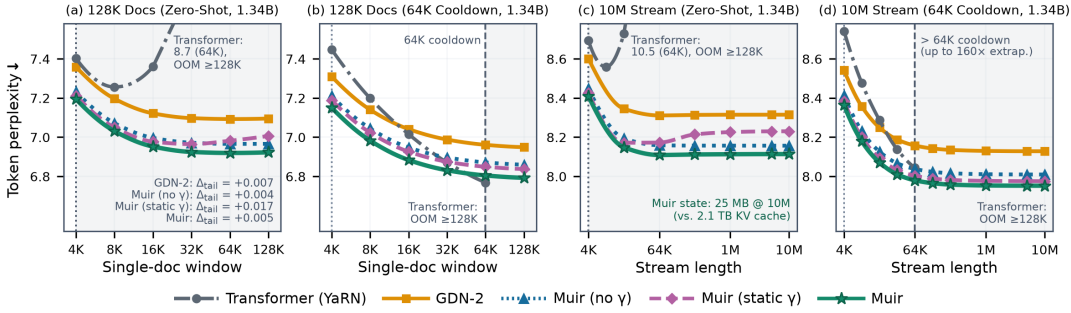
Dynamic readout improves multi-hop and multi-needle recall



- RULER six-task mean at 1K–4K: 73.0% Muir, 61.8% GDN-2, 70.9% Transformer.
- Five-hop Variable Tracking: 34.3% vs. 21.5% GDN-2 and 8.8% Transformer.
- Based/JRT 2K-cut mean: 34.60% vs. 32.08% GDN-2.

Paper, Sec. 5.2.2 / Fig. 6, PDF p. 17; Appendix H / Tables 13–14, pp. 52–53.

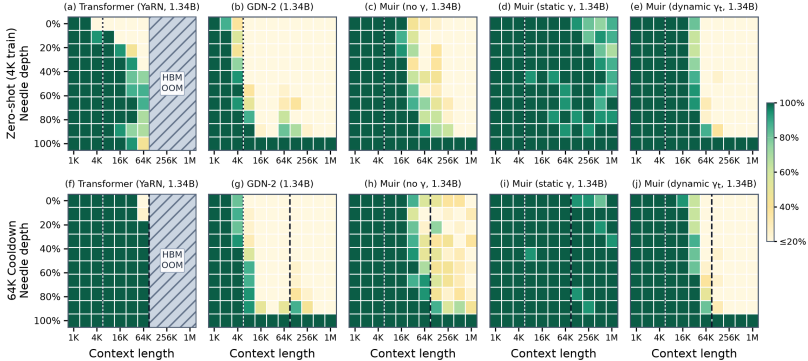
Dynamic readout keeps language-modeling loss stable on long streams



- Zero-shot from 4K, full Muir reaches 8.11 PPL at 10M tokens; GDN-2 is 8.31.
- The 24-layer fp32 recurrent state is 25 MB, independent of stream length.
- Static gain drifts on continuous text; a 64K cooldown recalibrates it.

Paper, Sec. 5.2.3 / Fig. 7, PDF p. 18; Appendix H / Table 16, p. 55.

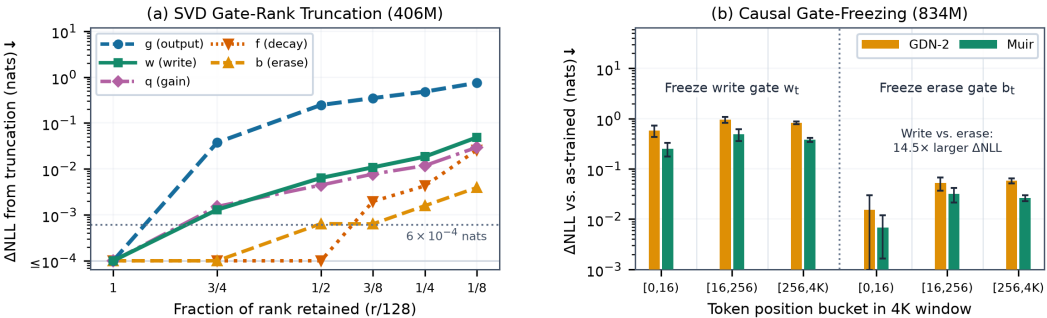
Static readout is better for a single passkey query



- Static-gain Muir reaches 90.0% zero-shot and 99.4% after cooldown at 1M tokens.
- Dynamic Muir trades write rank for lag-adaptive routing and tapers beyond 32K on this single-needle task.

Paper, Sec. 5.2.3 / Fig. 8, PDF p. 19; RULER S-NIAH-1, Appendix H / Table 17, p. 56.

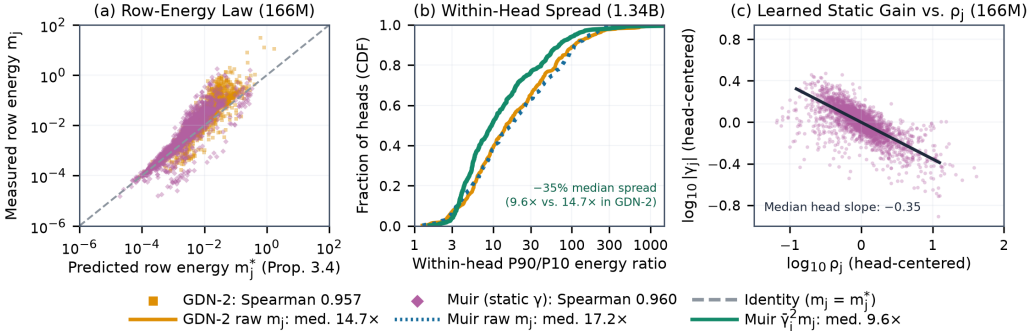
Trained models exhibit the predicted gate-sensitivity asymmetry



- SVD truncation: decay and erase saturate by rank 64; write, output, and query readout need rank 128.
- Freezing the write gate increases NLL 14.5× more than freezing erase.

Paper, Sec. 6 / Fig. 10, PDF p. 22; supplementary rank sweep in Appendix I / Table 19, p. 58.

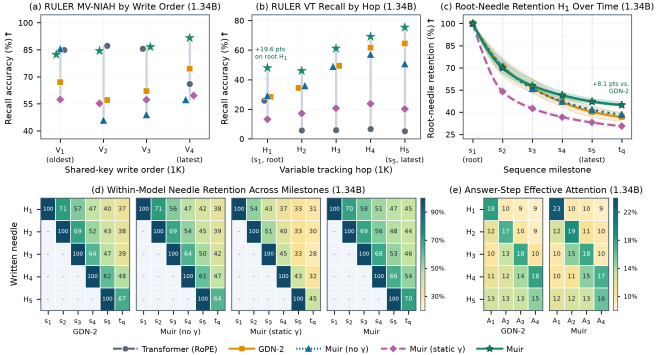
Readout gain compresses row-energy imbalance



- Gain-weighted p90/p10 row-energy spread: 9.6× in Muir vs. 14.7× in GDN-2.
- Learned static gains follow $\gamma_j \propto \rho_j^{-0.35}$, between unit gain and full whitening.
- Mean erase rate falls 16.6%, preserving longer-lived associations.

Paper, Sec. 6 / Fig. 11 and Table 3, PDF p. 23.

Dynamic gain routes readout across the hops of a binding chain



- Root binding retention at query: 37% in GDN-2, 45% in Muir.
- Dynamic gain shifts readout toward the target hop across answer steps.

Paper, Sec. 6 / Fig. 13, PDF p. 24; full write-order and hop-resolved results, Appendix I / Table 23, p. 61.

What the evidence supports, and where the design trades off

Supported by the reported experiments

- Sensitivity-aware gate ranks preserve quality and free FFN capacity.
- Matched-filter readout reduces row imbalance and write-time erasure.
- Boundary pivots enable stable large-chunk TPU training in sampled checkpoints.

Specific trade-offs

- Static gain: single-lag retrieval to 1M.
- Dynamic gain: multi-hop and 10M stream modeling.
- Transformer is faster below roughly 11K.
- Theory uses stationary and second-order approximations; training uses one seed.

Take-home: state capacity depends on how the model allocates write, erase, and readout control.

Paper, Secs. 5–7, PDF pp. 14–25; full identity: arXiv:2611.08376v1.