Probing Hybrid Language Models for In-Context Recall

Pol Puigdemont [1] **Volkan Cevher** [1]

Abstract

Hybrid language models that interleave quadratic-attention with linear-attention layers can match quadratic-attention models on in-context recall while being more efficient at long context, but how they actually handle recall inside the architecture is not yet understood. Recent mechanistic work locates the recall circuit at the prediction position via head ablations on a few publicly-released hybrids, where attention placement is fixed by the original training recipe. We pretrain placement-controlled hybrids at 340M alongside quadratic-attention and linear-attention baselines, and to see how they perform recall we construct the sequence-layer map, a 2D probe-accuracy matrix over layers and token positions in the residual stream that we apply uniformly across architectures. Attention placement determines whether a hybrid recalls, while standard language-modeling evaluations stay flat across placements. At the prediction position, the recall signal undergoes a sharp layer-localized phase transition only in hybrids able to recall and in quadratic-attention models. Across the sequence, quadratic-attention models drop the recall signal and look it up at the prediction position; hybrids inherit this pattern depending on the attention layer's position. A head-level analysis at 340M and on OLMo-Hybrid 7B further localizes the phase transition to a small set of attention heads, including some we can only see by reading what each head writes into the answer-token logits. Together, these results give a controlled view of how pretrained hybrid language models handle in-context recall, and why some succeed where others fail.

[1]EPFL, Lausanne, Switzerland. Correspondence to: Pol Puigdemont <polpuigdemont@gmail.com>.

Mechanistic Interpretability Workshop at the 43rd International Conference on Machine Learning, Seoul, South Korea, 2026.
Copyright 2026 by the author(s).

1. Introduction

Quadratic-attention has been the main primitive for sequence mixing in large language models since the transformer (Vaswani et al., 2017), but its key-value cache grows linearly with the sequence and its compute scales quadratically. Linear-attention models replace this cache with a recurrent state of bounded size, updated one token at a time, which keeps the memory budget fixed and the operation count linear in the sequence length (Katharopoulos et al., 2020). The expressivity cost of compressing the past into a bounded state is recall, the ability to retrieve a specific earlier token from the prompt; Arora et al. (2024a) show that this deficit drives most of the perplexity gap between linear-attention and quadratic-attention models. Hybrid models interleave quadratic-attention layers into otherwise linear-attention stacks and recover most of the recall at a fraction of the long-context cost (Merrill et al., 2026; Lieber et al., 2024), and a wave of recent production releases is adopting this recipe at scale. Yet why and how a few quadratic-attention layers recover recall is still not understood.

Of the design axes hybrids expose, we isolate the position of the quadratic-attention layer inside the stack. Theory shows that a single quadratic-attention layer suffices for in-context retrieval without specifying its depth (Wen et al., 2025), so we pretrain eleven 340M checkpoints on 15B SlimPajama tokens (Soboleva et al., 2023): six twenty-layer Gated DeltaNet (GDN) hybrids with one quadratic-attention layer at L0, L4, L8, L12, L16, or L19; two linear-attention controls, GDN and Gated Linear Attention (GLA); and three quadratic-attention controls, Multi-Head Attention (MHA), Grouped-Query Attention (GQA), and Multi-Head Latent Attention (MLA). Recall on the RULER long-context benchmark fans across a 59-point range as the layer is moved, while the same checkpoints differ by less than three points on a nine-task language-model evaluation (LM-eval) average (Figure 1). Hybrid comparisons at this scale typically rely on aggregate LM-eval averages, which miss this variation, so we use targeted recall to study placement (Hsieh et al., 2024).

We open the residual stream with three strands of analysis that share a single interface (Section 3). The first is a sequence-layer probe map of the residual at every (token-position, layer) cell, and it shows that placement of the sin-

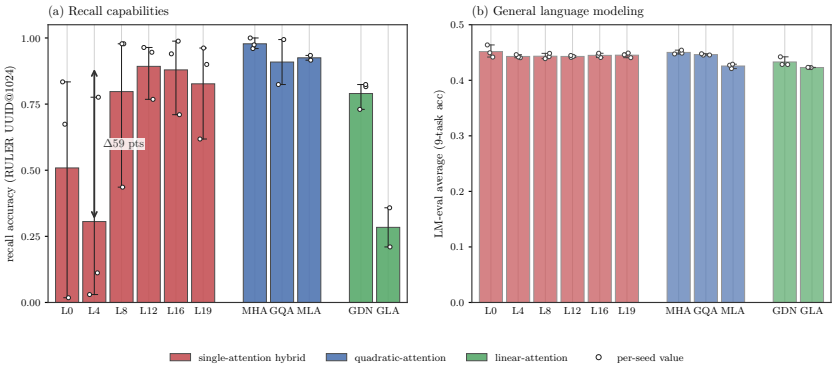

Figure 1. **A single attention layer's position determines RULER recall, yet 9-task LM-eval scores see none of it.** *(a)* RULER `niah_single_uuid@1024` recall fans across a 59-point range as a single quadratic-attention layer is moved through a 20-layer single-attention hybrid (L0–L19). linear-attention models (GDN, GLA) fall below the hybrid floor; quadratic-attention models (MHA, GQA, MLA) cluster near the hybrid ceiling (mean 0.94). *(b)* The same 11 models on a 9-task LM-eval average cluster within ∼3 points (0.42–0.45), so placement is invisible. Around 340M parameters, 15B SlimPajama tokens, $n = 2$–3 training seeds drawn from $\{1, 2, 42\}$; per-seed values shown as dots. Task list in Appendix A.7; per-task numbers in Appendix D.1.

gle quadratic-attention layer determines whether the recall target becomes linearly decodable at the prediction position (Section 4.1, Figure 5). The second is a head-level account that combines mean ablation with language-model (LM) head projection of the per-head residual write, and yields a four-class taxonomy on the 340M sweep and on the released OLMo-Hybrid 7B Base model (Section 4.2, Figure 6). One of the four classes, the *echo-writer* heads, is ablation-invariant yet projection-positive, so ablation alone cannot detect it; at 7B the taxonomy separates the heads that move recall accuracy under ablation from those that write the answer in projection; the two form disjoint head populations. The third is methodological: a rank-one cancellation in the gated delta rule (Section 4.3) makes a haystack of repeated keys a no-op on linear-attention memory while a haystack of distinct keys is not. Conclusions about whether linear-attention layers retain recall information therefore need to be reported across the synthetic-versus-text and repeat-versus-essay axes (Jelassi et al., 2024) rather than read off a single task.

Contributions. (a) A placement-controlled pretraining sweep at 340M that isolates the position of a single quadratic-attention layer in an otherwise linear-attention stack and shows the recall gap it opens is invisible to aggregate LM-eval averages. (b) A sequence-layer probe map of the residual stream that pinpoints where placement matters: the recall target reaches the prediction position cleanly only for late single-attention placements and pure-quadratic-attention controls, with the gap to early placements and pure-linear-attention controls sharpest on the more discriminating universally unique identifier (UUID) task. (c) A head-level account, architecture-agnostic across GDN, GLA, and quadratic-attention blocks, that distinguishes an *echo-writer* class of attention heads that ablation alone misses; the same four-class structure is read off both the 340M sweep and the released OLMo-Hybrid 7B Base model (Merrill et al., 2026), and at 7B the heads that affect recall under ablation and the answer-writing heads under projection occupy different layer populations (stability check across post-training stages in Appendix B.2). (d) A methodological reconsideration of recall evidence across synthetic-versus-text tasks and repeat-versus-essay fillers, with the filler asymmetry traced to a rank-one cancellation in the delta-rule update that turns repeated-key tokens into no-ops on the recurrent state.

2. Preliminaries

We treat each model as operating on a shared *residual stream* (Elhage et al., 2021). Tokens embed to $r_t^{(0)} \in \mathbb{R}^d$. A pre-norm layer ℓ (Vaswani et al., 2017; Xiong et al., 2020) writes twice into the residual: first through its sequence mixer, which mixes information across sequence positions (e.g. attention), then through its channel mixer, which acts position-wise across feature dimensions (e.g. an MLP). The sequence mixer reads $\mathrm{LN}(r^{(\ell)})$ at all positions $\leq t$, where LN denotes layer normalization, and writes a contribution $z_t^{(\ell)}$, producing an intermediate residual

$$\widetilde{r}_t^{(\ell)} \;=\; r_t^{(\ell)} + z_t^{(\ell)}. \tag{1}$$

The channel mixer then reads $\mathrm{LN}(\widetilde{r}_t^{(\ell)})$ and writes $\widetilde{z}_t^{(\ell)}$ back into the residual:

$$r_t^{(\ell+1)} \;=\; \widetilde{r}_t^{(\ell)} + \widetilde{z}_t^{(\ell)}. \tag{2}$$

The channel mixer is held fixed across all architectures we study. Hybrid models differ only in their sequence mixer, and that is where our analysis focuses. After L such layers, the LM-head produces token logits at position t as $W_U\,\mathrm{LN}\big(r_t^{(L)}\big) \in \mathbb{R}^{|V|}$, where each layer normalization has its own learned scale and shift, identified by its input, and $W_U \in \mathbb{R}^{|V|\times d}$ is the unembedding matrix.

For quadratic-attention layers, the sequence mixer is a causal attention mixer whose output is a weighted sum over past value vectors; MHA (Vaswani et al., 2017) and GQA (Ainslie et al., 2023) are the two instances used in the main text, and they share the per-head structure developed below. Writing $x_t = \mathrm{LN}(r_t^{(\ell)})$, head h projects x into queries, keys, and values,

$$q_t^{(h)} = W_q^{(h)} x_t, \qquad k_t^{(h)} = W_k^{(h)} x_t, \qquad v_t^{(h)} = W_v^{(h)} x_t, \tag{3}$$

and attends over the past to produce a per-head output

$$a_t^{(h)} = \sum_{s \le t} \mathrm{softmax}_s \left(\frac{\langle q_t^{(h)}, k_s^{(h)} \rangle}{\sqrt{d_h}} \right) v_s^{(h)}. \tag{4}$$

The sequence mixer's write at position t sums head outputs through their per-head projections,

$$z_t^{(\ell)} = \sum_h W_o^{(h)} a_t^{(h)}, \tag{5}$$

and we index attention heads as Lℓ-hh for head h of layer ℓ. Linear-attention layers replace the dot-product softmax with a bounded recurrent state S_t (Katharopoulos et al., 2020) that evolves as

$$S_t = f_t(S_{t-1}) + w_t(x_t), \tag{6}$$

where f_t gates or decays the previous state and w_t writes new information from the current token; the per-head structure of the write into the residual stream matches Equation (5), and the main-text instances are GDN (Yang et al., 2025) and GLA (Yang et al., 2024a). A hybrid model combines the two kinds of sequence mixer in the same residual stream. The main text focuses on the single-attention hybrid: one quadratic-attention layer at a chosen depth in an otherwise linear-attention stack. At this scale this minimal configuration already recovers most of the recall accuracy of richer hybrids, and it gives a clean comparison surface for sweeping over placement and the underlying linear-attention family.

A linear probe is a fixed readout fit to the residual stream of a frozen model (Alain & Bengio, 2016; Hewitt & Liang, 2019). Given a residual vector $r_t^{(\ell)}$ and a target $y \in \{1, \ldots, C\}$, we fit a multinomial-logistic readout $W_p \in \mathbb{R}^{d \times C}$ that produces class logits $W_p^\top r_t^{(\ell)}$. We use probes as a representational instrument and pair them with a behavioral one: recall accuracy is whether the model produces the correct answer when it decodes the target, while probe accuracy measures whether that same target is linearly decodable from a chosen residual cell. Reporting both separates what the model emits from what is linearly represented in its residual stream.

3. Method

Our probes target a single recall task: a RULER `niah_single` prompt (Hsieh et al., 2024; Kamradt, 2023) hides a target value inside otherwise unrelated filler text and asks the model to retrieve it on demand. Each prompt contains five named spans and four canonical anchors (POST-NEEDLE, POST-FILLER, PRE-AP, PREDICT); Figure 2 sketches the scaffold and concrete cards per filler regime are in Appendix A.1. We sweep two axes: the target is a 7-digit number (`niah_single_num`) or a UUID-style sequence (`niah_single_uuid`), and the filler is a repeating template haystack (*repeat* filler) or held-out Paul-Graham essay text (*essay* filler). The filler axis changes the task rather than merely its presentation: it interacts non-trivially with delta-rule memory in ways that change what linear-attention layers can or cannot retain through the haystack (algebra in Section 4.3). We keep both regimes in scope below and report across the full 2×2 task-by-filler grid.

We fit a linear probe (Section 2) independently at every cell (t, ℓ), taking as input the post-layer residual $r_t^{(\ell+1)}$ and as label y the index of the queried needle keyword among a fixed candidate pool. Each fit reports held-out accuracy from stratified five-fold cross-validation averaged across probe seeds, and the cells assemble into a probe map M indexed by (t, ℓ) (Figure 3). We read M at two granularities: a depth curve at each of the four anchors, or a position-by-depth sequence-layer map on a stride-spaced ladder of absolute positions $t_0, t_{32}, \ldots$ that exposes where in the sequence the target is linearly represented.

Per-head ablation, zero and mean. To identify which heads affect recall accuracy, we measure how the model's behaviour changes when a single head's write is removed from the residual stream. The sequence mixer write at layer ℓ is a sum over heads (Equation (5)). We replace head h's contribution at every position t either by zero or by its calibration-set mean $\bar{a}^{(h)}$, and read the resulting drop in recall accuracy:

$$\Delta \mathrm{R}_\bullet^{(h)} = \mathrm{R} - \mathrm{R}\big(a_t^{(h)} \leftarrow c_\bullet^{(h)}\big), \qquad c_{\text{zero}}^{(h)} = 0, \quad c_{\text{mean}}^{(h)} = \bar{a}^{(h)}, \tag{7}$$

where $\bar{a}^{(h)} = \mathbb{E}_{(p,t) \in \mathcal{D}_{\text{cal}}}\big[a_t^{(h)}\big]$ is taken over a calibration set $\mathcal{D}_{\text{cal}}$ of *essay* filler prompts at all positions, with $n = 30$ prompts at $L = 1024$ for the 340M corpus and $n = 100$ prompts at $L = 512$ for OLMo-Hybrid, all at a single training seed (`seed=42`). A large positive $\Delta \mathrm{R}_{\text{mean}}^{(h)}$ means the head's typical write strongly contributes to recall; a value near zero means it does not. Zero ablation, the standard form in prior work (Wu et al., 2025; Bick et al., 2025), conflates the head's causal contribution with a residual-norm shift, because zeroing $a_t^{(h)}$ removes mass the model has

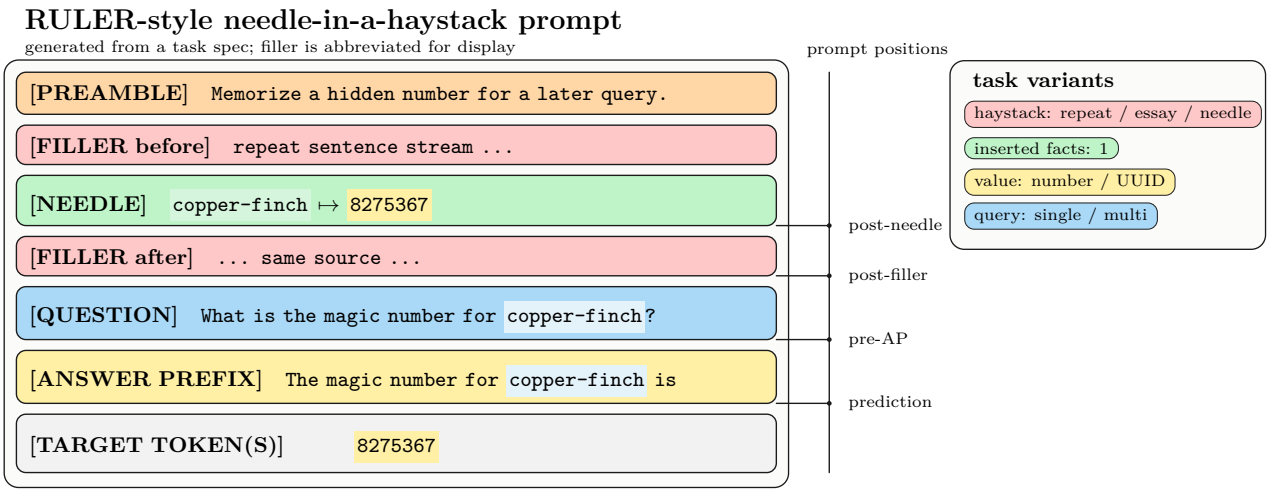

Figure 2. **Needle-in-a-haystack task structure.** The prompt contains a preamble, length-controlled filler, one or more keyword-value needles, a query, an answer prefix, and the target value. Context length is controlled separately by the filler budget.

not been trained against. Mean ablation is the calibrated counterfactual: it preserves the typical write magnitude and isolates information loss.

LM-head projection of the per-head residual write. To identify what each head writes into the answer direction (which ablation cannot see), we measure the head's per-position contribution in vocabulary space. We project the per-head residual contribution through the same final layer norm and unembedding W_U that the LM-head applies (Section 2):

$$\pi_t^{(h)} \;=\; \mathrm{LN}\big(W_o^{(h)}\, a_t^{(h)}\big)\, W_U^{\top} \in \mathbb{R}^{|V|}. \tag{8}$$

The top tokens of $\pi_t^{(h)}$ are the head's vote on the next-token distribution from its slice of the residual stream alone, isolated from every other head and channel. We summarise this vote two ways, one for what the head writes consistently across prompts and one for whether the recall target lands in its top tokens on a given prompt. The template view averages top-20 ranks of $\pi_t^{(h)}$ across the calibration set and isolates constant writers whose top tokens are the same on every prompt, typically colons or wordpiece prefixes. The per-prompt windowed rate $\mathrm{rate}^{(h)}_{\max\,K}$, defined next, measures whether $y(p)$ enters the head's top-20 within $K = 5$ free-decode positions on prompt p:

$$\mathrm{rate}^{(h)}_{\max\,K} = \mathbb{E}_{p\in\mathcal{D}_{\mathrm{cal}}}\Big[\max_{t\in W_K(p)} \mathbf{1}\Big[y(p) \in \mathrm{top}_{20}\big(\pi_t^{(h)}\big)\Big]\Big], \tag{9}$$

where $W_K(p)$ are the first $K = 5$ free-decode positions on prompt p. A high $\mathrm{rate}^{(h)}_{\max\,5}$ flags heads that emit their content one token after the colon, which a position-locked summary would miss. The two views are nearly disjoint on the non-silent heads of our corpus: a template-view hit calls a constant writer, a rate-view hit calls a content writer, and the partition we define next (Table 1) inherits this separation.

Head categories. Combining each head's mean-ablation magnitude $|\Delta\mathrm{R}^{(h)}_{\mathrm{mean}}|$ with its windowed content rate $\mathrm{rate}^{(h)}_{\max\,5}$ partitions the population into four categories at two thresholds, $\varepsilon_\Delta = 0.05$ on $|\Delta\mathrm{R}|$ and $r_{\mathrm{thr}} = 0.5$ on $\mathrm{rate}_{\max\,5}$. Table 1 states the rule and the one-line reading of each cell. A fifth marker *no-Δ-data* covers heads with no ablation row available, currently the MHA layers of OLMo-Hybrid Base.

This partition applies to quadratic-attention and linear-attention heads alike: both families write through a per-head $W_o^{(h)}$ slice of $z_t^{(\ell)}$ (Section 2), so projection and ablation run head-for-head on either. Sensitivity to the threshold choice $(\varepsilon_\Delta, r_{\mathrm{thr}})$ is reported in the appendix.

4. Probing hybrids for in-context recall

Setup. The body corpus is 11 checkpoints pretrained at 340M parameters on 15B SlimPajama tokens with a single matched recipe (Appendix A.2): six single-quadratic-attention placements at $\ell \in \{0, 4, 8, 12, 16, 19\}$ on a 20-layer linear-attention backbone, plus pure-linear-attention controls (GDN, GLA) and pure-quadratic-attention controls (MHA, GQA, MLA) at the same parameter band. The head-level analysis extends to the released OLMo-Hybrid 7B Base model (Merrill et al., 2026) as a scale-and-realism check. Evaluations use RULER `niah_single` (Hsieh et al., 2024) restricted to `num` and `uuid` at $L = 1024$, $n = 300$ prompts per cell, with both filler regimes *repeat* filler and *essay* filler kept on every body plot so the filler axis

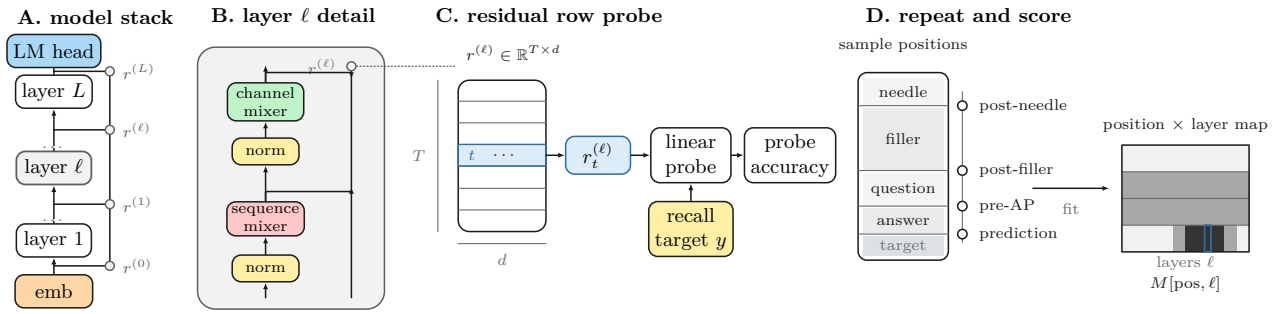

Figure 3. **Residual-stream probing methodology.** For each input sequence, we read the post-layer residual state $r^{(\ell+1)} \in \mathbb{R}^{T\times d}$ after the full layer update, i.e. after both the sequence mixer and channel mixer have contributed to the residual stream. At a selected token position t, the probe receives the row vector $r_t^{(\ell+1)}$ and predicts the recall-target class y with a linear classifier. Scores are held-out linear-probe accuracies from stratified five-fold cross-validation, averaged over probe seeds. Repeating this over layers and token positions gives the probe map $M[\text{pos}, \ell]$. The current anchor figures use the four named positions POST-NEEDLE, POST-FILLER, PRE-AP, and PREDICT; the broader heatmap pipeline also adds stride-spaced absolute positions $t_0, t_{32}, \ldots$ for $T = 1024$.

Table 1. **Four-class head taxonomy from the two-threshold rule.** The *echo-writer* cell motivates pairing the two tools: projection recovers a category that ablation alone misses, because an ablation-invariant head can still write the recall target into the answer direction.

Category	Threshold rule	Interpretation
answer-writer	$\lvert\Delta\mathrm{R}^{(h)}_{\text{mean}}\rvert > \varepsilon_\Delta \;\wedge\; \mathrm{rate}^{(h)}_{\max 5} \geq r_{\text{thr}}$	writes the recall target and affects recall under ablation
echo-writer	$\lvert\Delta\mathrm{R}^{(h)}_{\text{mean}}\rvert \leq \varepsilon_\Delta \;\wedge\; \mathrm{rate}^{(h)}_{\max 5} \geq r_{\text{thr}}$	writes the recall target but ablation is silent
structural	$\lvert\Delta\mathrm{R}^{(h)}_{\text{mean}}\rvert > \varepsilon_\Delta \;\wedge\; \mathrm{rate}^{(h)}_{\max 5} < r_{\text{thr}}$	affects recall under ablation without writing answer content
silent	$\lvert\Delta\mathrm{R}^{(h)}_{\text{mean}}\rvert \leq \varepsilon_\Delta \;\wedge\; \mathrm{rate}^{(h)}_{\max 5} < r_{\text{thr}}$	neither writes the target nor moves recall accuracy

stays explicit. The placement-controlled corpus is trained at seeds $\{1, 2, 42\}$; bars and tiles report the median, with per-seed values overlaid as dots. Per-seed disaggregations are in Appendix D; protocol pins in Appendix A.6.

4.1. How hybrids do recall

The placement effect of Figure 1 fans recall accuracy across sixty points but does not say where in the prompt or at what depth the recall target enters the residual stream; reading probe accuracy at every (token-position, depth) cell of the residual shows both axes at once. We collapse the surface two ways. Figure 4 reads the depth curve at the prediction position for each architecture, and the placement of the single quadratic-attention layer determines how strongly the recall target is represented at the prediction position. Late placements (L8, L12, L16) and pure-quadratic-attention controls show a sharp layer-localised phase transition: probe accuracy is near zero up to a depth that depends on the model and near one above it. Early placements (L0, L4) and pure-linear-attention controls show a much shallower crossing, and the gap to late placements is sharpest on the more discriminating `niah_single_uuid` task; on `niah_single_num` the early placements still reach mid-range probe accuracy at the prediction position.

Figure 5 keeps the full position-by-depth surface as a heatmap on the same checkpoints. Quadratic-attention controls and late-placement hybrids show a wedge that opens at the prediction position late in depth, with the late-layer crossing pinned to the placement of the single quadratic-attention layer. Pure-linear-attention controls and early-placement hybrids reach the prediction position much more weakly, with the gap most visible on `niah_single_uuid`; the POST-NEEDLE column is the only anchor at which linear-attention models decode the target reliably across depth, consistent with the *essay* filler regime stressing the recurrent state through the haystack.

4.2. Head-level results

We apply the head-level tools of Section 3 to the 71 attention heads of the 340M body corpus and the 240 heads of the eight quadratic-attention layers $\{3, 7, 11, 15, 19, 23, 27, 31\}$ in OLMo-Hybrid Base; full head counts and protocol pins in Appendix A.6. Figure 6 plots each 340M head on the signed mean-ablation against projection-rate plane and partitions them into the four cells of Table 1 at ε_Δ=0.05 and r_{thr}=0.5. Four heads land in the *echo-writer* panel: they project the recall target into the answer direction yet leave recall accuracy unmoved when ablated, a category an ablation-only methodology cannot detect. The four *answer-writer* heads and the four *echo-writer* heads are all single-attention place-

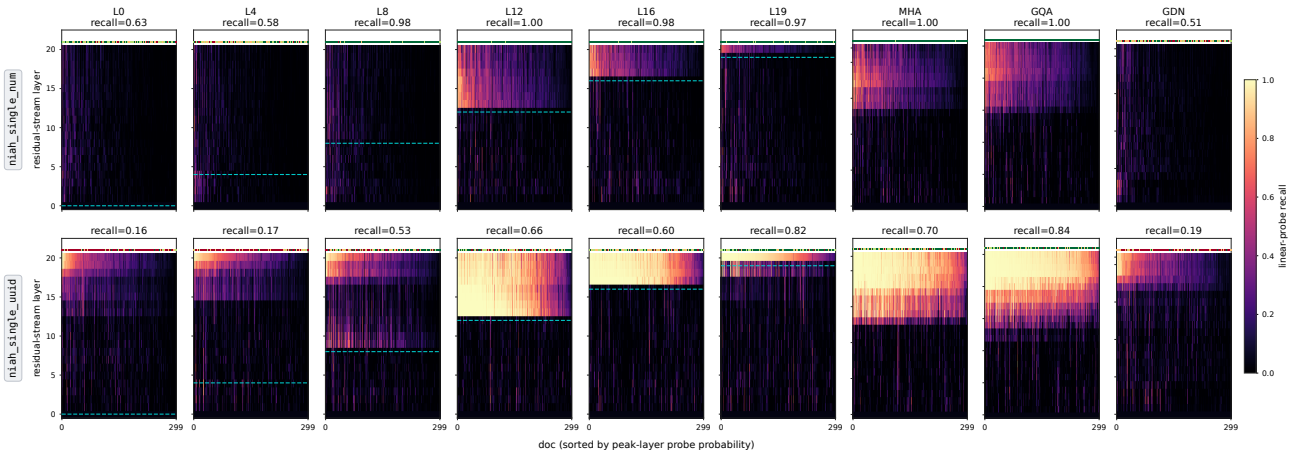

Figure 4. **A phase transition in residual-stream recall emerges as a single quadratic-attention layer moves through the hybrid stack.** Per-doc linear-probe recall: y = residual-stream layer, x = doc index sorted by peak-layer probability; *(a)* `niah_single_num`, *(b)* `niah_single_uuid`. The cyan dashed line marks the quadratic-attention layer in each single-attention hybrid. Columns are family-grouped: hybrids L0–L19, then quadratic-attention controls (MHA, GQA), then a linear-attention control (GDN). RULER niah_single@1024, *essay* filler. Protocol, behavioural-correctness strip, and per-seed coverage in Appendices A.6 and D.

ments at deeper layers; the six *structural* heads are entirely linear-attention controls. The partition is robust to threshold choice (Appendix B).

Applying the same four-class partition to OLMo-Hybrid Base under colors-essay (a single-token regime, since the released checkpoint cannot reproduce multi-token RULER answers) uncovers a regime shift (Appendix B.3). Of the 26 heads on the zero-ablation shortlist, 23 (88%) become *silent* under mean ablation: a head that hurts recall accuracy when zeroed out need not hurt it when its slice is replaced with the calibration mean, a confound we revisit in Section 4.3. The remaining three are *structural* (top: L15-h7 at $|\Delta|$=0.11). The heads that actually write the answer are absent from the shortlist at near-zero ablation drop and cluster on two of the eight quadratic-attention layers: L23-h1 projects `blue` at $\mathrm{rate}_{\max 5}$=0.99, L27-h4 projects `violet` at 0.67, L23-h18 projects `azure` at 0.61, and L27-h5 belongs to the same cluster on the broader answer-side template at rate 0.72. At 7B the heads identified by ablation and the answer-writing heads identified by projection occupy different layer populations; these heads appear only along the projection axis.

A cross-stage check confirms the head population identified by mean ablation is not Base-specific: under the original repeat-filler regime, the ablation-ranked top-K bundle (L11-h$\{1, 9, 13\}$+L7-h4) remains sensitive to ablation across OLMo-Hybrid's post-training stages and the head identities do not reshuffle (Appendix B.2).

Reconciliations. Wu et al. (2025) identify retrieval heads in dense transformers via attention-mass-on-needle and recover them as the *answer-writer* special case of the broader four-class structure; their attention-pattern primitive misses the *echo-writer* class on OLMo-Hybrid (L23-h1 writes `blue` at $\mathrm{rate}_{\max 5}$=0.99 yet does not move recall accuracy when ablated). Bick et al. (2025) use per-head zero ablation on Zamba2 to delegate the aggregate role of recall to attention layers in natively-pretrained hybrids; the OLMo-Hybrid regime shift sharpens this to a finding zero ablation alone misses, that the heads that affect recall under ablation and the heads that write the answer need not be the same. Michalak & Abreu (2025)'s behaviour-level finding that linear-attention layers contribute zero retrieval at the prediction position is consistent with our observation that the heads carrying recall mass are attention heads.

4.3. Revisiting recall evidence

Two methodological checks complement the head-level analysis (Figure 7). Panel (a) plots ΔR at OLMo-Hybrid Base under zero versus mean ablation: every cumulative bundle lies above the $y{=}x$ diagonal, with the gap reaching 0.10 to 0.25 for the late-layer cluster (L11-h$\{1, 9, 13\}$+L7-h4, the same heads that anchor Appendix B.2). Reporting mean ablation, as Section 4.2 adopts, removes the residual-norm shock zero ablation conflates with causal contribution. Panel (b) reads the residual probe along the prompt-position axis for GDN and MHA under both filler regimes: the needle remains decodable from GDN's POST-FILLER residual under *repeat* filler but not under *essay* filler, while MHA stays layer-localised to prediction position either way.

A short calculation specialised to GDN's sequence mixer makes the asymmetry mechanistic. Specialising the abstract linear-attention recurrence of Equation (6) to the gated delta rule (Schlag et al., 2021; Yang et al., 2025):

$$S_t \;=\; \alpha_t\, S_{t-1}\big(I - \beta_t\, k_t k_t^\top\big) \;+\; \beta_t\, v_t\, k_t^\top, \qquad (10)$$

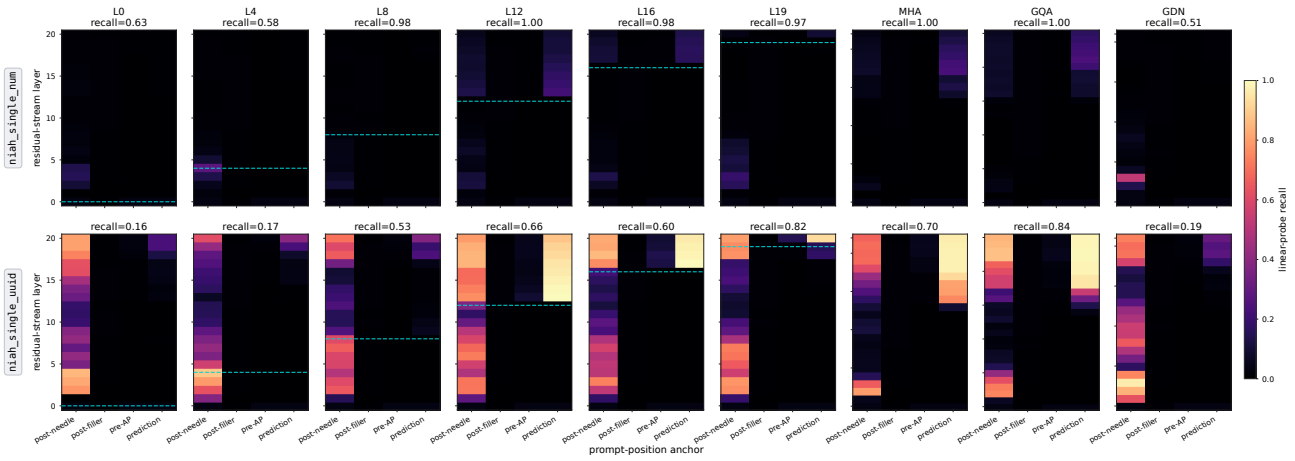

Figure 5. **The same phase transition recast along the prompt's token axis: quadratic-attention layers and successful hybrids preserve recall content through to the prediction position, linear-attention models do not.** sequence-layer map per model: y = residual-stream layer, x = the four canonical anchors (POST-NEEDLE, POST-FILLER, PRE-AP, prediction position); *(a)* `niah_single_num`, *(b)* `niah_single_uuid`. The cyan dashed line marks the quadratic-attention layer in each single-attention hybrid. Columns share Figure 4's family-grouped order. Protocol and per-seed coverage in Appendices A.6 and D.

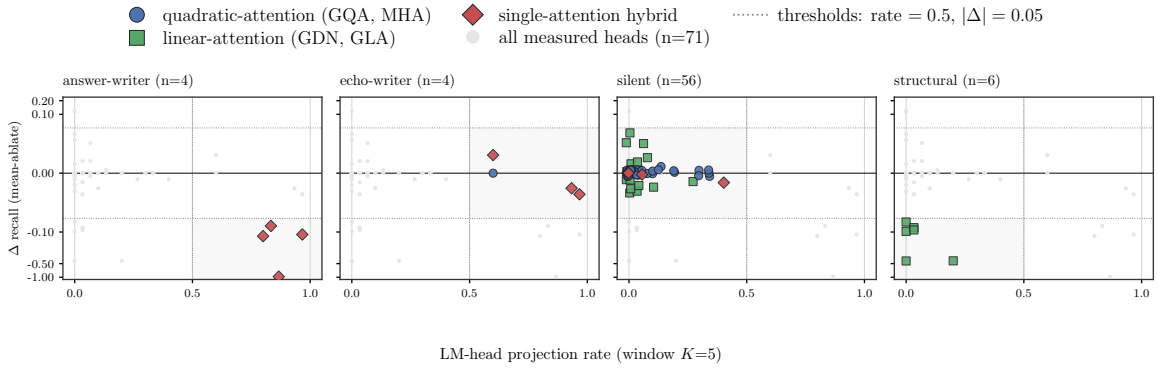

Figure 6. **Four heads project the recall target into the answer direction yet leave recall accuracy unmoved under mean ablation, the *echo-writer* class that an ablation-only methodology cannot detect.** Each of the n=71 attention heads of the 340M body corpus is placed in one of four panels by the threshold rule of Table 1: x-axis is the windowed projection rate $\mathrm{rate}^{(h)}_{\max 5}$, y-axis is mean-ablation $\Delta\mathrm{R}^{(h)}_{\mathrm{mean}}$. Every panel shows the full population as small grey dots and colours only the heads in its own category, by architecture class. Protocol in Appendix A.6. Panels show 70 of the 71 heads; the single polluting head ($\Delta\mathrm{R}^{(h)}_{\mathrm{mean}} > 0$, where ablation helps recall) appears only as a background dot. The y-axis is symlog (linear within $|\Delta\mathrm{R}| \leq 0.05$).

where k_t, v_t are the per-token key/value projections, $\beta_t \in (0,1)$ the write strength, and $\alpha_t \in (0,1)$ the scalar forget gate. The transition $(I - \beta_t k_t k_t^\top)$ is a generalised Householder operator (Yang et al., 2024b; Grazzi et al., 2025), with eigenvalue 1 on $k_t^\perp$ and $1 - \beta_t \|k_t\|^2$ along k_t.

Repeated keys are no-ops. Set $\alpha_t = 1$ (the un-gated delta rule) and suppose the model wrote (k, v) at some earlier step so that $S_{t-1}k = v$. At any later step with $k_t = k$ and $v_t = v$, expanding Equation (10) gives

$$\begin{aligned} S_t &= S_{t-1} \;-\; \beta_t\,(S_{t-1}k)\,k^\top \;+\; \beta_t\,v\,k^\top \\ &= S_{t-1} \;-\; \beta_t\,v\,k^\top \;+\; \beta_t\,v\,k^\top \;=\; S_{t-1}. \end{aligned} \tag{11}$$

The rank-one erase $\beta_t\,(S_{t-1}k)\,k^\top$ cancels the rank-one write $\beta_t\,v\,k^\top$ identically, independent of β_t: a token whose (k, v) has already been fitted leaves the state unchanged. With the gate restored ($\alpha_t < 1$) the same expansion gives $S_t = \alpha_t\,S_{t-1} + (1-\alpha_t)\beta_t\,v\,k^\top$ and $S_t k = v\,[\alpha_t + (1-\alpha_t)\beta_t\|k\|^2]$, a partial *refresh* of the (k, v) slot on top of the gate's uniform contraction. The same fixed point falls out of the test-time-regression view of Wang et al. (2025b): the delta rule is one step of stochastic gradient descent on $\frac{1}{2}\|Sk - v\|^2$, whose gradient vanishes once the example is fitted. Siems et al. (2025, Prop. 1.1) state the same collapse (a product of identical-key Householders is a single Householder with rescaled β) as a degenerate end of their expressivity argument; here it is the source of the methodological asymmetry between filler regimes.

RULER's repeated-boilerplate variant of `niah_single` (Hsieh et al., 2024) repeats a fixed

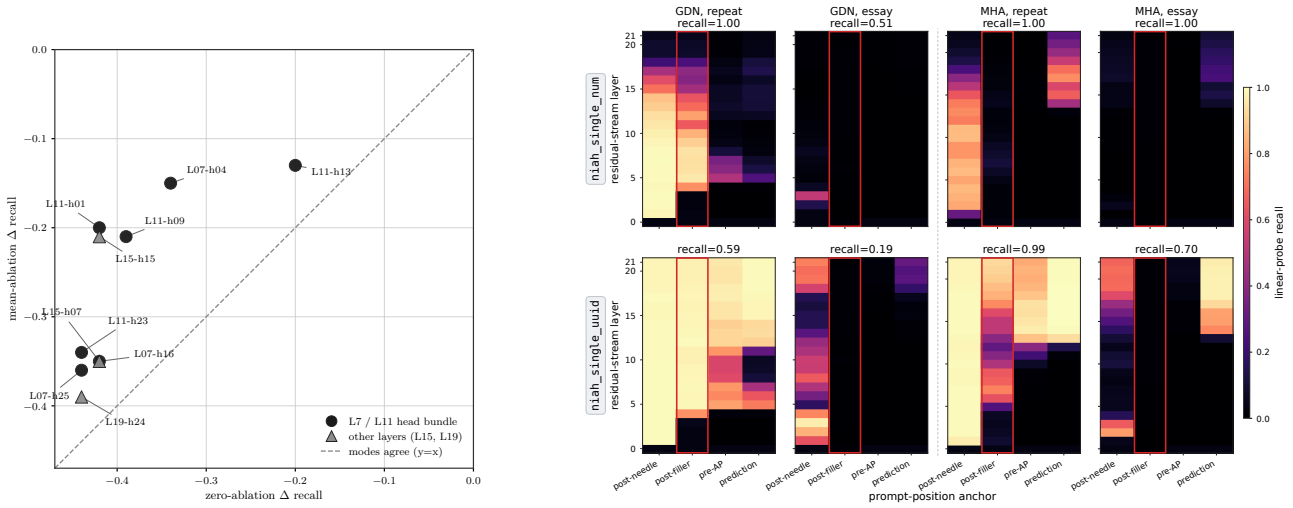

Figure 7. **(a) Zero ablation overstates head importance: every cumulative bundle lies above the $y{=}x$ diagonal.** Top-10 cumulative head bundles at OLMo-Hybrid Base, ΔR under zero ablation (x) versus mean ablation (y). Perpendicular distance to the diagonal is the residual-norm shock zero ablation conflates with causal contribution; for the L11-h$\{1, 9, 13\}$+L7-h4 cluster the gap reaches 0.10 to 0.25. Bundle composition and protocol in Appendix A.6. **(b) Filler regime, not architecture, gates whether recall content survives the haystack.** sequence-layer map for GDN and MHA crossed with *repeat* filler and *essay* filler; columns in order GDN *repeat* filler, GDN *essay* filler, MHA *repeat* filler, MHA *essay* filler; rows top `niah_single_num`, bottom `niah_single_uuid`. Red rectangle marks POST-FILLER, the diagnostic anchor. The within-architecture filler contrast is decisive: GDN's POST-FILLER residual survives *repeat* filler (col 1) and goes dark under *essay* filler (col 2); MHA stays layer-localised to prediction position either way. Mechanism in Equation (11); protocol in Appendix A.6.

sentence to length, so after its first occurrence every subsequent filler token *refreshes* the needle's (k, v) slot rather than perturbing it: the readout $S_t k_n$ is preserved up to the scalar $[\alpha_t + (1 - \alpha_t)\beta_t \|k\|^2]$ at every step. *essay* filler replaces this with held-out Paul-Graham essay text where each filler token has a distinct key roughly orthogonal to the needle's slot; the writes consume capacity elsewhere in S, and the needle's slot, no longer refreshed, decays in expectation as $\prod_t \alpha_t$ across L filler steps. The two regimes ask different questions of the model, and only the second is the regime under which a delta-rule recurrent memory must actually compete with attention. Natural-text haystacks have been advocated before on naturalness and distributional-plausibility grounds (Kuratov et al., 2024); the calculation above gives a mechanistic reason specific to delta-rule recurrent models. Figure 7(b) confirms the prediction directly: GDN's POST-FILLER residual survives *repeat* filler but collapses under *essay* filler; MHA is unaffected by the swap. A recall scalar averaged across either filler would conflate the two regimes; the sequence-layer map separates them.

5. Conclusion

A placement-controlled pretraining sweep, a residual-stream sequence-layer map, and a head-level account that pairs mean ablation with LM-head projection let us read how natively-pretrained hybrids handle in-context recall. The paper has three main findings. Aggregate language-modelling averages stay flat over a sixty-point range in targeted recall as a single quadratic-attention layer is moved through the stack. The residual probe shows that placement decides whether the target reaches the prediction position at all, with the gap to early placements and pure-linear-attention controls sharpest on UUID. On OLMo-Hybrid 7B Base, the four-class head taxonomy exposes a regime shift in which the heads identified by ablation and the answer-writing heads identified by projection occupy different layer populations (Appendix B.3).

The diagnostic kit (sequence-layer map probe, per-head zero and mean ablation, LM-head projection of the per-head write, plus the *repeat* filler versus *essay* filler stress test) is reusable. The present 340M and 7B corpus does not settle multi-needle and multi-hop recall with the same head-level analysis, longer contexts where the linear-attention state is forced to compete, or head-level interleaved hybrids whose recall circuit may live where the present sweep does not place attention; one specific causal question, the L1 linear-attention mixer's contribution to the near-perfect POST-NEEDLE probe in OLMo-Hybrid versus the embedding and L0 writes that share the same residual, is flagged in Appendix C and we leave it to a causal companion that pairs the probe with an L1 attention-output ablation.

References

Ainslie, J., Lee-Thorp, J., de Jong, M., Zemlyanskiy, Y., Lebrón, F., and Sanghai, S. GQA: Training generalized multi-query transformer models from multi-head checkpoints. In *Proceedings of EMNLP*, 2023. URL https://arxiv.org/abs/2305.13245. arXiv:2305.13245.

Airlangga, M. C., AlQuabeh, H., Nwadike, M. S., and Inui, K. Emergence of primacy and recency effect in mamba: A mechanistic point of view, 2025. URL https://arxiv.org/abs/2506.15156. arXiv:2506.15156.

Alain, G. and Bengio, Y. Understanding intermediate layers using linear classifier probes, 2016. arXiv:1610.01644.

Ali, A., Zimerman, I., and Wolf, L. The hidden attention of Mamba models, 2024. URL https://arxiv.org/abs/2403.01590. arXiv:2403.01590.

Arora, A., Rathi, N., Selvam, N. R., Csordás, R., Jurafsky, D., and Potts, C. Mechanistic evaluation of transformers and state space models. *arXiv preprint arXiv:2505.15105*, 2025. URL https://arxiv.org/abs/2505.15105.

Arora, S., Eyuboglu, S., Timalsina, A., Johnson, I., Poli, M., Zou, J., Rudra, A., and Ré, C. Zoology: Measuring and improving recall in efficient language models. In *ICLR*, 2024a.

Arora, S., Eyuboglu, S., Zhang, M., Timalsina, A., Alberti, S., Zinsley, D., Zou, J., Rudra, A., and Ré, C. Simple linear attention language models balance the recall–throughput tradeoff. In *ICML*, 2024b.

Bick, A., Xing, E. P., and Gu, A. Understanding the skill gap in recurrent language models: The role of the gather-and-aggregate mechanism. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025. URL https://arxiv.org/abs/2504.18574. arXiv:2504.18574.

Bick, A., Xing, E. P., and Gu, A. Retrieval-aware distillation for transformer-SSM hybrids, 2026. URL https://arxiv.org/abs/2602.11374. arXiv:2602.11374.

Bisk, Y., Zellers, R., Le Bras, R., Gao, J., and Choi, Y. PIQA: Reasoning about physical commonsense in natural language. In *AAAI*, 2020. URL https://arxiv.org/abs/1911.11641. arXiv:1911.11641.

Borobia, H., Seguí-Mas, E., and Tormo-Carbó, G. Functional component ablation reveals specialization patterns in hybrid language model architectures, 2026. URL https://arxiv.org/abs/2603.22473. arXiv:2603.22473.

Botev, A., De, S., Smith, S. L., et al. Recurrentgemma: Moving past transformers for efficient open language models, 2024. URL https://arxiv.org/abs/2404.07839. arXiv:2404.07839.

Clark, C., Lee, K., Chang, M.-W., Kwiatkowski, T., Collins, M., and Toutanova, K. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *NAACL*, 2019. URL https://arxiv.org/abs/1905.10044. arXiv:1905.10044.

Clark, P., Cowhey, I., Etzioni, O., Khot, T., Sabharwal, A., Schoenick, C., and Tafjord, O. Think you have solved question answering? try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018. URL https://arxiv.org/abs/1803.05457.

De, S., Smith, S. L., Fernando, A., Botev, A., Cristian-Muraru, G., Gu, A., Haroun, R., Berrada, L., Chen, Y., Srinivasan, S., Desjardins, G., Doucet, A., Budden, D., Teh, Y. W., Pascanu, R., De Freitas, N., and Gulcehre, C. Griffin: Mixing gated linear recurrences with local attention for efficient language models, 2024. URL https://arxiv.org/abs/2402.19427. arXiv:2402.19427. Hawk and Griffin.

DeepSeek-AI. DeepSeek-V2: A strong, economical, and efficient mixture-of-experts language model, 2024. URL https://arxiv.org/abs/2405.04434. arXiv:2405.04434.

Dong, X., Fu, Y., Diao, S., Byeon, W., Chen, Z., Mahabaleshwarkar, A. S., Liu, S.-Y., Van Keirsbilck, M., Chen, M.-H., Suhara, Y., Lin, Y., Kautz, J., and Molchanov, P. Hymba: A hybrid-head architecture for small language models. In *ICLR*, 2025. URL https://arxiv.org/abs/2411.13676. arXiv:2411.13676.

Elhage, N., Nanda, N., Olsson, C., Henighan, T., Joseph, N., Mann, B., Askell, A., Bai, Y., Chen, A., Conerly, T., DasSarma, N., Drain, D., Ganguli, D., Hatfield-Dodds, Z., Hernandez, D., Jones, A., Kernion, J., Lovitt, L., Ndousse, K., Amodei, D., Brown, T., Clark, J., Kaplan, J., McCandlish, S., and Olah, C. A mathematical framework for transformer circuits, 2021. URL https://transformer-circuits.pub/2021/framework/index.html. Transformer Circuits Thread, Anthropic.

Endy, N., Grosbard, I. D., Ran-Milo, Y., Slutzky, Y., Tshuva, I., and Giryes, R. Mamba knockout for unraveling factual information flow. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2025. URL https://aclanthology.org/2025.acl-long.1143/. arXiv:2505.24244.

Ensign, D. and Garriga-Alonso, A. Investigating the indirect object identification circuit in Mamba. In *ICML Workshop on Mechanistic Interpretability*, 2024. URL https://arxiv.org/abs/2407.14008. arXiv:2407.14008.

Gao, L., Tow, J., Abbasi, B., Biderman, S., Black, S., DiPofi, A., Foster, C., Golding, L., Hsu, J., Le Noac'h, A., Li, H., McDonell, K., Muennighoff, N., Ociepa, C., Phang, J., Reynolds, L., Schoelkopf, H., Skowron, A., Sutawika, L., Tang, E., Thite, A., Wang, B., Wang, K., and Zou, A. The language model evaluation harness. 2024. URL https://github.com/EleutherAI/lm-evaluation-harness. EleutherAI evaluation harness.

Glorioso, P., Anthony, Q., Tokpanov, Y., Golubeva, A., Shyam, V., Whittington, J., Pilault, J., and Millidge, B. The Zamba2 suite: Technical report, 2024a. URL https://arxiv.org/abs/2411.15242. arXiv:2411.15242.

Glorioso, P., Anthony, Q., Tokpanov, Y., Whittington, J., Pilault, J., Ibrahim, A., and Millidge, B. Zamba: A compact 7B SSM hybrid model, 2024b. URL https://arxiv.org/abs/2405.16712. arXiv:2405.16712.

Grazzi, R., Siems, J., Zela, A., Franke, J. K. H., Hutter, F., and Pontil, M. Unlocking state-tracking in linear RNNs through negative eigenvalues. In *ICLR*, 2025. URL https://arxiv.org/abs/2411.12537. arXiv:2411.12537.

Hewitt, J. and Liang, P. Designing and interpreting probes with control tasks. In *Proceedings of EMNLP-IJCNLP*, 2019. URL https://arxiv.org/abs/1909.03368. arXiv:1909.03368.

Hsieh, C.-P., Sun, S., Kriman, S., Acharya, S., Rekesh, D., Jia, F., Zhang, Y., and Ginsburg, B. Ruler: What's the real context size of your long-context language models? In *COLM*, 2024.

IBM Research. IBM granite 4.0: Hyper-efficient, high performance hybrid models for enterprise, 2025. URL https://www.ibm.com/new/announcements/ibm-granite-4-0-hyper-efficient-high-performance-hybrid-models. Model release.

Jelassi, S., Brandfonbrener, D., Kakade, S. M., and Malach, E. Repeat after me: Transformers are better than state space models at copying. In *ICML*, 2024. URL https://arxiv.org/abs/2402.01032. arXiv:2402.01032.

Kamradt, G. Pressure testing GPT-4-128k with long context recall, 2023. URL https://github.com/gkamradt/LLMTest_NeedleInAHaystack. Public benchmark release: "Needle in a Haystack".

Katharopoulos, A., Vyas, A., Pappas, N., and Fleuret, F. Transformers are RNNs: Fast autoregressive transformers with linear attention. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020. URL https://arxiv.org/abs/2006.16236. arXiv:2006.16236.

Kuratov, Y., Bulatov, A., Anokhin, P., Rodkin, I., Sorokin, D., Sorokin, A., and Burtsev, M. BABILong: Testing the limits of LLMs with long context reasoning-in-a-haystack, 2024. URL https://arxiv.org/abs/2406.10149. arXiv:2406.10149.

Li, Y., Yang, S., Tan, S., Mishra, M., Panda, R., Zhou, J., and Kim, Y. Distilling to hybrid attention models via KL-guided layer selection, 2025. URL https://arxiv.org/abs/2512.20569. arXiv:2512.20569.

Lieber, O., Lenz, B., Bata, H., Cohen, G., Osin, J., Dalmedigos, I., Safahi, E., Meirom, S., Belinkov, Y., Shalev-Shwartz, S., Abend, O., Alon, R., Asida, T., Bergman, A., Glozman, R., Gokhman, M., Manevich, A., Ratner, N., Rozen, N., Shwartz, E., Zusman, M., and Shoham, Y. Jamba: A hybrid transformer-Mamba language model, 2024. URL https://arxiv.org/abs/2403.19887. arXiv:2403.19887.

M-A-P Team. GDN-1.3B-100B-Pure: Open reproduction of Gated DeltaNet, 2024. URL https://huggingface.co/m-a-p/1.3B-100B-GatedDeltaNet-pure. Hugging Face model card.

Merrill, W., Petty, J., and Sabharwal, A. The illusion of state in state-space models. In *ICML*, 2024. URL https://arxiv.org/abs/2404.08819. arXiv:2404.08819.

Merrill, W., Li, Y., Romero, T., Svete, A., Costello, C., Dasigi, P., Groeneveld, D., Heineman, D., Kuehl, B., Lambert, N., Li, C., Lo, K., Malik, S., Matusz, D., Minixhofer, B., Morrison, J., Soldaini, L., Timbers, F., Walsh, P., Smith, N. A., Hajishirzi, H., and Sabharwal, A. Olmo hybrid: From theory to practice and back, 2026. URL https://arxiv.org/abs/2604.03444. arXiv:2604.03444.

Michalak, F. and Abreu, S. Some attention is all you need for retrieval. In *NeurIPS Workshop on Mechanistic Interpretability*, 2025. URL https://arxiv.org/abs/2510.19861. arXiv:2510.19861.

Mihaylov, T., Clark, P., Khot, T., and Sabharwal, A. Can a suit of armor conduct electricity? A new dataset for open book question answering. In *EMNLP*,

2018. URL https://arxiv.org/abs/1809.02789. arXiv:1809.02789.

MiniMax Team. MiniMax-01: Scaling foundation models with lightning attention, 2025. URL https://arxiv.org/abs/2501.08313. arXiv:2501.08313.

NVIDIA. Nemotron-H: A family of accurate and efficient hybrid Mamba-Transformer models, 2025. URL https://arxiv.org/abs/2504.03624. arXiv:2504.03624.

Okpekpe, D. and Orvieto, A. Revisiting associative recall in modern recurrent models, 2025. URL https://arxiv.org/abs/2508.19029. arXiv:2508.19029.

Olsson, C., Elhage, N., Nanda, N., Joseph, N., DasSarma, N., Henighan, T., Mann, B., Askell, A., Bai, Y., Chen, A., Conerly, T., Drain, D., Ganguli, D., Hatfield-Dodds, Z., Hernandez, D., Johnston, S., Jones, A., Kernion, J., Lovitt, L., Ndousse, K., Amodei, D., Brown, T., Clark, J., Kaplan, J., McCandlish, S., and Olah, C. In-context learning and induction heads, 2022. URL https://arxiv.org/abs/2209.11895. Anthropic technical report. arXiv:2209.11895.

Paperno, D., Kruszewski, G., Lazaridou, A., Pham, Q. N., Bernardi, R., Pezzelle, S., Baroni, M., Boleda, G., and Fernández, R. The LAMBADA dataset: Word prediction requiring a broad discourse context. In *ACL*, 2016. URL https://arxiv.org/abs/1606.06031. arXiv:1606.06031.

Paulo, G., Marshall, T., and Belrose, N. Does transformer interpretability transfer to RNNs? In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025. URL https://arxiv.org/abs/2404.05971. arXiv:2404.05971.

Qwen Team. Qwen3-next: Towards ultimate training & inference efficiency, 2025. URL https://qwen.ai/blog?id=4074cca80393150c248e508aa62983f9cb7d27cd. Model release.

Ren, L., Liu, Y., Lu, Y., Shen, Y., Liang, C., and Chen, W. Samba: Simple hybrid state space models for efficient unlimited context language modeling, 2024. URL https://arxiv.org/abs/2406.07522. arXiv:2406.07522.

Sakaguchi, K., Le Bras, R., Bhagavatula, C., and Choi, Y. WinoGrande: An adversarial Winograd schema challenge at scale. In *AAAI*, 2020. URL https://arxiv.org/abs/1907.10641. arXiv:1907.10641.

Schlag, I., Irie, K., and Schmidhuber, J. Linear transformers are secretly fast weight programmers. In *ICML*, 2021. URL https://arxiv.org/abs/2102.11174. arXiv:2102.11174.

Sharma, A. S., Atkinson, D., and Bau, D. Locating and editing factual associations in mamba. In *Conference on Language Modeling (COLM)*, 2024. URL https://arxiv.org/abs/2404.03646. arXiv:2404.03646.

Siems, J., Carstensen, T., Zela, A., Hutter, F., Pontil, M., and Grazzi, R. DeltaProduct: Improving state-tracking in linear RNNs via householder products. In *ICLR*, 2025. URL https://arxiv.org/abs/2502.10297. arXiv:2502.10297.

Singh, A. K., Moskovitz, T., Hill, F., Chan, S. C. Y., and Saxe, A. M. What needs to go right for an induction head? a mechanistic study of in-context learning circuits and their formation. In *ICML*, 2024. URL https://arxiv.org/abs/2404.07129. arXiv:2404.07129.

Soboleva, D., Al-Khateeb, F., Myers, R., Steeves, J., Hestness, J., and Dey, N. SlimPajama: A 627B token cleaned and deduplicated version of RedPajama, 2023. URL https://huggingface.co/datasets/cerebras/SlimPajama-627B. Cerebras dataset release.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *NeurIPS*, 2017.

Waleffe, R., Byeon, W., Riach, D., Norick, B., Korthikanti, V., Dao, T., Gu, A., Hatamizadeh, A., Singh, S., Narayanan, D., Kulshreshtha, G., Singh, V., Casper, J., Kautz, J., Shoeybi, M., and Catanzaro, B. An empirical study of Mamba-based language models, 2024. URL https://arxiv.org/abs/2406.07887. arXiv:2406.07887.

Wang, D., Zhu, R.-J., Abreu, S., Shan, Y., Kergan, T., Pan, Y., Chou, Y., Li, Z., Zhang, G., Huang, W., and Eshraghian, J. A systematic analysis of hybrid linear attention. *arXiv preprint arXiv:2507.06457*, 2025a. URL https://arxiv.org/abs/2507.06457. arXiv:2507.06457.

Wang, J., Ge, X., Shu, W., Tang, Q., Zhou, Y., He, Z., and Qiu, X. Towards universality: Studying mechanistic similarity across language model architectures, 2024. URL https://arxiv.org/abs/2410.06672. arXiv:2410.06672.

Wang, K. A., Shi, J., and Fox, E. B. Test-time regression: A unifying framework for designing sequence models with associative memory, 2025b. URL https://arxiv.org/abs/2501.12352. arXiv:2501.12352.

Welbl, J., Liu, N. F., and Gardner, M. Crowdsourcing multiple choice science questions. In *Workshop on Noisy User-generated Text (W-NUT)*, 2017. URL https://arxiv.org/abs/1707.06209. arXiv:1707.06209.

Wen, K., Dang, X., and Lyu, K. RNNs are not Transformers (yet): The key bottleneck on in-context retrieval. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://arxiv.org/abs/2402.18510. arXiv:2402.18510.

Wu, W., Wang, Y., Xiao, G., Peng, H., and Fu, Y. Retrieval head mechanistically explains long-context factuality. In *ICLR (Oral)*, 2025. URL https://arxiv.org/abs/2404.15574. arXiv:2404.15574.

Xiong, R., Yang, Y., He, D., Zheng, K., Zheng, S., Xing, C., Zhang, H., Lan, Y., Wang, L., and Liu, T.-Y. On layer normalization in the transformer architecture. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020. URL https://arxiv.org/abs/2002.04745. arXiv:2002.04745.

Yang, S. and Zhang, Y. Fla: A triton-based library for hardware-efficient implementations of linear attention mechanism, January 2024. URL https://github.com/fla-org/flash-linear-attention.

Yang, S., Wang, B., Shen, Y., Panda, R., and Kim, Y. Gated linear attention transformers with hardware-efficient training. In *ICML*, 2024a.

Yang, S., Wang, B., Zhang, Y., Shen, Y., and Kim, Y. Parallelizing linear transformers with the delta rule over sequence length. In *NeurIPS*, 2024b. URL https://arxiv.org/abs/2406.06484. arXiv:2406.06484.

Yang, S., Kautz, J., and Hatamizadeh, A. Gated delta networks: Improving mamba2 with delta rule. In *ICLR*, 2025.

Zellers, R., Holtzman, A., Bisk, Y., Farhadi, A., and Choi, Y. HellaSwag: Can a machine really finish your sentence? In *ACL*, 2019. URL https://arxiv.org/abs/1905.07830. arXiv:1905.07830.

Zuo, J., Velikanov, M., Chahed, I., Belkada, Y., Rhayem, D. E., Kunsch, G., Hacid, H., Yous, H., Farhat, B., Khadraoui, I., et al. Falcon-H1: A family of hybrid-head language models redefining efficiency and performance, 2025. URL https://arxiv.org/abs/2507.22448. arXiv:2507.22448.

A. Methodology details

A.1. Data examples for each setting

The recall tasks are drawn from the RULER suite (Hsieh et al., 2024), restricted to two single-needle variants: `niah_single_1` (a 7-digit number) and `niah_single_3` (a UUID built as three 4-character alphanumeric blocks). Each prompt is anchored at four canonical positions used throughout the paper: POST-NEEDLE immediately after the needle is written, POST-FILLER after the filler block ends, PRE-AP just before the answer prefix, and PREDICT the position at which the model emits the answer. The primary filler regime in this appendix is *essay* filler, built from the Paul Graham essay corpus and its 500k-character offset variant, since this is the regime where placement effects can be separated from surface-level repetition cues. The *repeat* filler foil reuses a 5-sentence haystack template and runs alongside *essay* filler in the body figures. The body sketch of the scaffold is Figure 2; Figure 8 pins one concrete prompt per (task, filler) cell with the four anchors marked.

A.2. Pretraining recipe

The placement-controlled corpus is trained under a single matched recipe so that every cross-architecture comparison runs at fixed compute, fixed data ordering, and fixed tokeniser. Architectural depth is the only quantity that varies, and its variation is held to the narrow band needed to keep the parameter budget on target (Appendix A.4). Optimiser, schedule, and dataloader settings follow the public GLA 1.3B release (Yang et al., 2024a) via the Flash Linear Attention (FLA) reference implementation (Yang & Zhang, 2024); the recipe defaults are offloaded to those upstream references and only the numbers that pin our specific run are listed here.

The pinned settings are: 28 610 optimiser steps for $\approx 1.500 \times 10^{10}$ tokens (15B to four significant figures, identical across the corpus); a global batch of 256 sequences of length 2048; AdamW with $\beta_1 = 0.9$, $\beta_2 = 0.95$, $\varepsilon = 10^{-15}$; a cosine schedule with 1024 warmup steps and peak learning rate 4×10^{-4}; bf16 mixed precision with fp32 reductions; default training seed 42; the SlimPajama-627B release (Soboleva et al., 2023) sampled through a stratified 59B-token manifest; and the LlamaTokenizer artifact published with the GLA 1.3B release (vocabulary size 32 000), so every architecture sees identical token IDs in identical order. We deviate from the upstream recipe on no other axis.

A.3. Per-architecture configuration

The body corpus is locked at eleven canonical checkpoints, listed in Table 2. The table is read together with the recipe of Appendix A.2: every row shares $d_{\text{model}} = 1024$, SwiGLU MLPs, RMSNorm, untied input and output embeddings, and the matched optimiser and data ordering above. What differs is depth and the placement, count, and shape of the attention layer.

The eleven rows are organised into three families. The first two are the linear-attention baselines GDN and GLA, which fix the linear-recurrent-only end of the spectrum. The next six rows are the single-attention hybrids, all sharing a 20-layer Qwen3-Next backbone (Qwen Team, 2025) in which one linear-attention block at position $\ell \in \{0, 4, 8, 12, 16, 19\}$ is replaced by a MHA block with 4 query heads and one key-value (KV) head; this is the placement-controlled axis the paper interrogates. The final three rows are the quadratic-attention controls MHA, GQA, and MLA, which complete the corpus by isolating attention shape (full multi-head, grouped-query, latent-KV) from the placement question. We deliberately keep depth as the single hyperparameter we are willing to vary. The single-attention family inherits the 20-layer Qwen3-Next backbone unchanged, so all six placements compare at identical depth. The pure baselines need a small depth adjustment to land near the 340M target at fixed $d_{\text{model}} = 1024$, since GDN, GLA, MHA, and GQA have different per-layer parameter footprints. Holding the recipe constant and letting depth absorb the parameter-budget delta is more conservative than re-tuning learning rate or batch size per architecture, and it keeps the comparison dimension that the paper cares about (placement and family) from being entangled with optimisation choices.

CONFIGURATION SOURCES

Architecture configurations follow the FLA reference implementations (Yang & Zhang, 2024) for every row of Table 2. The single-attention family also adopts the Qwen3-Next (Qwen Team, 2025) hybrid layout that FLA's `gated_deltanet` module exposes through its `attn` dictionary.

Deviations from the FLA defaults are mechanical and limited to four. (i) The GDN baseline runs in the V1 GDN variant, not the V2 default; the V1 fallback keeps the per-layer parameter footprint inside the 340–380M band at $d_{\text{model}} = 1024$.

color key: stored keyword queried keyword target value K = inserted facts
multi-needle prompts interleave sampled filler chunks with each inserted needle, then add trailing filler before the question

niah_single_1 repeat haystack, number value — haystack: repeat | value: number | K=1 | query: single

[PREAMBLE] A special magic number is hidden within the following text. Make sure to memorize it. I will quiz you about the number afterwards.
[FILLER before] The grass is green. The sky is blue. The sun is yellow. Here we go. There and back again. ...
[NEEDLE] One of the special magic numbers for copper-finch is: 8275367.
[FILLER after] ... same source to the requested length ...
[QUESTION] What is the special magic number for copper-finch mentioned in the provided text?
[ANSWER PREFIX] The special magic number for copper-finch mentioned in the provided text is
[TARGET] 8275367

niah_single_2 essay haystack, number value — haystack: essay | value: number | K=1 | query: single

[PREAMBLE] A special magic number is hidden within the following text. Make sure to memorize it. I will quiz you about the number afterwards.
[FILLER before] Natural-language prose from a held-out essay stream continues here, with the needle inserted at a sampled depth. ...
[NEEDLE] One of the special magic numbers for pewter-anvil is: 8015764.
[FILLER after] ... same source to the requested length ...
[QUESTION] What is the special magic number for pewter-anvil mentioned in the provided text?
[ANSWER PREFIX] The special magic number for pewter-anvil mentioned in the provided text is
[TARGET] 8015764

niah_single_3 essay haystack, UUID value — haystack: essay | value: UUID | K=1 | query: single

[PREAMBLE] A special magic number is hidden within the following text. Make sure to memorize it. I will quiz you about the number afterwards.
[FILLER before] Natural-language prose from a held-out essay stream continues here, with the needle inserted at a sampled depth. ...
[NEEDLE] One of the special magic numbers for obsidian-quilt is: epf9-1dho-dzdo.
[FILLER after] ... same source to the requested length ...
[QUESTION] What is the special magic number for obsidian-quilt mentioned in the provided text?
[ANSWER PREFIX] The special magic number for obsidian-quilt mentioned in the provided text is
[TARGET] epf9-1dho-dzdo

niah_multikey_1 four keys in context, one queried key — haystack: essay | value: number | K=4 | query: single

[PREAMBLE] A special magic number is hidden within the following text. Make sure to memorize it. I will quiz you about the number afterwards.
[FILLER 0] sampled filler chunk before the first needle ...
[NEEDLE 1] One of the special magic numbers for woolen-astro is: 1781527.
[FILLER 1] ... sampled filler chunk between needles ...
[NEEDLE 2] One of the special magic numbers for thorny-boulder is: 3234302.
[FILLER 2] ... sampled filler chunk between needles ...
[NEEDLE 3] One of the special magic numbers for molten-gavel is: 5858837.
[FILLER 3] ... sampled filler chunk between needles ...
[NEEDLE 4] One of the special magic numbers for rustic-lantern is: 8031986.
[FILLER 4] ... trailing filler before the question ...
[QUESTION] What is the special magic number for molten-gavel mentioned in the provided text?
[ANSWER PREFIX] The special magic number for molten-gavel mentioned in the provided text is
[TARGET] 5858837

niah_multiquery four queried keys, all values must be generated — haystack: essay | value: number | K=4 | query: multi

[PREAMBLE] A special magic number is hidden within the following text. Make sure to memorize it. I will quiz you about the number afterwards.
[FILLER 0] sampled filler chunk before the first needle ...
[NEEDLE 1] One of the special magic numbers for coppice-orbit is: 3420198.
[FILLER 1] ... sampled filler chunk between needles ...
[NEEDLE 2] One of the special magic numbers for granite-pulse is: 2976225.
[FILLER 2] ... sampled filler chunk between needles ...
[NEEDLE 3] One of the special magic numbers for brittle-monsoon is: 6175466.
[FILLER 3] ... sampled filler chunk between needles ...
[NEEDLE 4] One of the special magic numbers for velvet-muon is: 4032085.
[FILLER 4] ... trailing filler before the question ...
[QUESTION] What are all the special magic numbers for coppice-orbit, granite-pulse, brittle-monsoon, velvet-muon mentioned in the provided text?
[ANSWER PREFIX] The special magic numbers for coppice-orbit, granite-pulse, brittle-monsoon, velvet-muon mentioned in the provided text are
[TARGET] 3420198, 2976225, 6175466, 4032085

Figure 8. **Appendix examples for RULER-style recall.** Each card is generated from the same task schema, with the queried keyword and expected value highlighted. Filler text is abbreviated only for display; context length is swept separately via the generator budget rather than shown as a per-card variant.

Table 2. **The eleven canonical checkpoints of the placement-controlled corpus.** All share $d_{\text{model}} = 1024$, SwiGLU, RMSNorm, untied embeddings, and the matched recipe of Appendix A.2. Single-attention placements share the Qwen3-Next 20-layer backbone and differ only in the position of the inserted MHA block; pure baselines vary depth slightly to land near the 340M parameter budget. Configuration sources and the FLA-default deviations are listed in Appendix A.3.

Architecture	Layers	Attention layers	Attention shape
GDN	21	0	n/a
GLA	24	0	n/a
single-attn $\ell = 0$	20	1 (L0)	4 heads / 1 KV head
single-attn $\ell = 4$	20	1 (L4)	4 heads / 1 KV head
single-attn $\ell = 8$	20	1 (L8)	4 heads / 1 KV head
single-attn $\ell = 12$	20	1 (L12)	4 heads / 1 KV head
single-attn $\ell = 16$	20	1 (L16)	4 heads / 1 KV head
single-attn $\ell = 19$	20	1 (L19)	4 heads / 1 KV head
MHA	22	22	8 heads / 8 KV heads
GQA	25	25	4 heads / 1 KV head
MLA	20	20	latent KV

(ii) The single-attention backbone runs GDN V2 in the Qwen3-Next layout, with the MHA block inserted at one of $\{\ell = 0, 4, 8, 12, 16, 19\}$ via FLA's `attn` dictionary; the only thing that varies across the six placement checkpoints is the index ℓ. (iii) The MHA, GQA, and MLA controls enable the same modernised attention features used inside the hybrids (query-key normalization, output gate, zero-centered gamma) so that the attention block is identical apart from head shape and KV-grouping. (iv) Width is set to $d_{\text{model}} = 1024$ across the corpus; depth follows Table 2. GLA otherwise runs against FLA defaults verbatim.

A.4. Parameter-matching note

The 340M parameter target drifts to roughly 340–380M across rows of Table 2. The drift is the price of holding $d_{\text{model}} = 1024$, SwiGLU expansion ratio, and RMSNorm placement fixed across families: at fixed width and matched MLPs, only depth is left to absorb the per-layer parameter difference between linear-recurrent and quadratic-attention sequence mixers. We accept the resulting $\pm 10\%$ band rather than re-tune width or expansion per architecture, because re-tuning would introduce a confound exactly along the dimension a fair comparison should hold constant. Per-architecture parameter counts follow mechanically from the depth column of Table 2 and the shared backbone settings; we do not list them separately.

A.5. Reproducibility

Each evaluation cell, indexed by (model, task, filler, sequence length), uses $n = 300$ prompts uniformly. Sequence length is 1024 in the primary results and 2048 as a secondary check, both inside the 2048-token training window. The two recall tasks are `niah_single_1` (7-digit number, slug `num`) and `niah_single_3` (UUID as three 4-character alphanumeric blocks, slug `uuid`); the two filler regimes are *repeat* filler (haystack of 5 sentences repeated) and *essay* filler (Paul Graham essays plus a 500k-character offset variant). Scoring is substring exact-match without normalization, matching the official RULER scorer. Decoding is greedy argmax with `max_new_tokens` of 10 for `num` and 16 for `uuid`; the wider budget for UUIDs is justified by the colon-tic side-finding in Appendix B.4. Probes are trained with 5 probe seeds and stratified 5-fold cross-validation per probe seed, yielding accuracy tensors of shape $(n_{\text{probe seeds}}, n_{\text{layers}}, n_{\text{folds}})$. The training-seed set used for multi-seed reporting is $\{1, 2, 42\}$.

A.6. Sampling and seed protocol

Cells and bars across body and appendix figures use the median across training seeds, with per-seed values overlaid as white-fill dots and a min–max bracket so that dispersion is visible without obscuring the median. The training-seed set is $\{1, 2, 42\}$ for every architecture in the corpus. A handful of (architecture, filler) eval cells finished after the submission cutoff and render as gray "no run" tiles in the per-seed grids of Appendix D; we do not impute or pad. Body figure captions report the seed range as n_{train}=2–3 for the same reason.

A.7. LM-eval protocol and task list

The "9-task LM-eval average" reported in Figure 1 and elsewhere in Section 1 is an arithmetic mean of zero-shot accuracy on the nine tasks listed in Table 3, all run through the EleutherAI `lm-evaluation-harness` (Gao et al., 2024). The metric reported per task is the harness's `acc,none` variant, so `hellaswag`, `piqa`, `arc_challenge`, `arc_easy`, and `openbookqa` use unnormalised accuracy rather than `acc_norm`; this matches the figure renderer cell-for-cell. The same harness invocation also writes results for two further tasks, `wikitext` and `copa`; we drop `wikitext` because it reports word-perplexity rather than accuracy and would distort an unweighted mean, and we drop `copa` because its 100-prompt validation split is noisy enough at 340M scale to dominate the cross-architecture spread. The remaining nine all use a $[0, 1]$ accuracy scale, are stable at this scale, and average without normalisation. Per-task numbers per checkpoint are tabulated in Table 5 and broken out per training seed in Table 6.

Table 3. **The nine tasks included in the "9-task LM-eval" average.** Zero-shot, EleutherAI `lm-evaluation-harness` (Gao et al., 2024), metric `acc,none` for every task. `wikitext` (perplexity, different scale) and `copa` (small split, noisy) are computed by the same harness invocation but excluded from the average.

Harness task	Metric	What it measures
`arc_challenge` (Clark et al., 2018)	`acc`	Grade-school multiple-choice science (challenge split)
`arc_easy` (Clark et al., 2018)	`acc`	Grade-school multiple-choice science (easy split)
`boolq` (Clark et al., 2019)	`acc`	Yes/no reading comprehension over short passages
`hellaswag` (Zellers et al., 2019)	`acc`	Sentence-continuation commonsense
`lambada_openai` (Paperno et al., 2016)	`acc`	Last-word prediction over narrative passages
`openbookqa` (Mihaylov et al., 2018)	`acc`	Multiple-choice elementary science with an open "book" of facts
`piqa` (Bisk et al., 2020)	`acc`	Physical commonsense reasoning
`sciq` (Welbl et al., 2017)	`acc`	Multiple-choice science exam questions
`winogrande` (Sakaguchi et al., 2020)	`acc`	Pronoun-coreference commonsense (large variant)

A.8. Context-length choice

The trained context length is 2048 and the primary evaluation length is 1024, with 2048 retained as a secondary check. Both lie inside the trained window, so any change in recall accuracy across architectures or placements is attributable to in-context retrieval rather than length-generalization behaviour outside training. Length generalization is out of scope for this paper. The 1024-token primary length still threads many tokens of filler between needle and answer prefix, so the recall task is not trivialized; it stays clear of the regime where architectures begin to differ purely on extrapolation.

A.9. Limitations

The recall task family is RULER `niah_single` restricted to the `num` and `uuid` variants; multi-needle, multi-hop, and natural long-context tasks are not exercised. The primary evaluation length is 1024 tokens (with 2048 as a secondary check), both inside the trained window, so the conclusions do not transfer mechanically to length-extrapolation regimes. The placement-controlled corpus is at 340M parameters and the head-level scale-up is the released OLMo-Hybrid Base hybrid; the four-class taxonomy is populated at both, but absolute populations of *answer-writer* and *echo-writer* are corpus-specific. The cross-stage trajectory across OLMo-Hybrid's post-training stages (Appendix B.2) is a stability check on a single model family rather than a cross-model claim. The reconciliation with H1 of Michalak & Abreu (2025) is currently representational: probe accuracy near 1 on the residual at depth $\ell{=}1$ at the POST-NEEDLE anchor in OLMo-Hybrid aggregates the embedding together with the L0 and L1 writes; pairing the same probe with an L1 attention-output ablation would isolate the mixer's contribution, and we leave it to future work. Architectures explored during prototyping but excluded from the canonical 11-checkpoint corpus are out of scope for the body claims.

A.10. Use of language models

The pretrained hybrid checkpoints are themselves the subjects of analysis, not part of the methodology. No language model is used as an important, original, or non-standard component of the core methods.

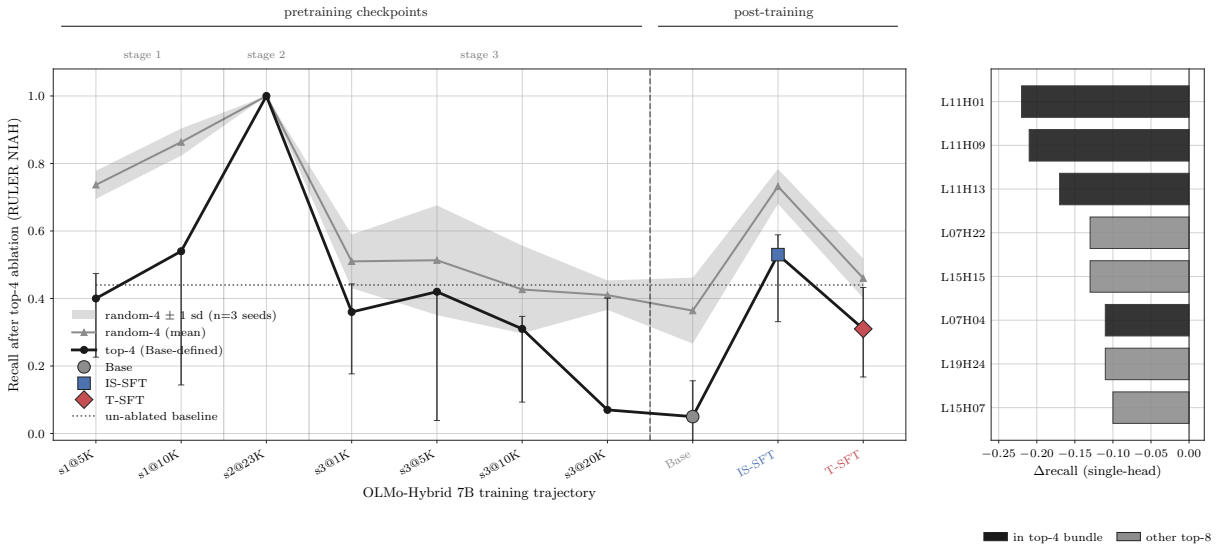

Figure 9. **The OLMo-Hybrid top-K head bundle is identified during Stage 3 and persists through SFT.** Left panel: recall accuracy after ablating the top-4 head bundle (focus colour, ranked by Base mean-ablation ΔR) versus a random size-matched bundle (gray, mean and ± 1 sd over n=3 random seeds), as a function of training stage. X-axis spans Stage 1 (`s1@{5K,10K}`), Stage 2 (`s2@23K`), Stage 3 (`s3@{1,5,10,20}K`), and the three post-training checkpoints (Base, Instruct-SFT, Think-SFT). The dotted line at ≈ 0.44 is the un-ablated Base recall accuracy. Right panel: the eight heads composing the Stage 3 top-K ranking, sorted by their single-head ΔR, with the top-4 bundle highlighted. At Stage 1 the random-K bundle is above top-K simply because the model is recall-naive; from Stage 3 onward the top-4 bundle is consistently more damaging to remove than random-4, and the identities (L11-h$\{1, 9, 13\}$ plus L7-h4) do not reshuffle through Instruct-SFT or Think-SFT. Error bars on the top-4 line are over n=3 probe seeds (the bundle itself is deterministic). K=4, L=512, RULER `niah_multikey` prompt set.

B. Additional experiments

B.1. Gate perturbation under filler swap

A natural follow-up to the no-op argument in Section 4.3 is to ask whether the prediction generalises beyond the all-ones gate. When the gate α_t is perturbed away from 1, the refresh argument behind Equation (11) relaxes in a controlled way: the per-token retention factor $[\alpha_t + (1 - \alpha_t)\beta_t \|k\|^2]$ acquired at every *repeat* filler step is replaced by a strict decay $\prod_t \alpha_t$ under *essay* filler, where filler keys are roughly orthogonal to the needle's slot. The body figure Figure 7(b) already exposes the asymmetry empirically: the needle remains decodable from GDN's POST-FILLER residual under *repeat* filler but not under *essay* filler, while MHA is invariant to the swap. We do not stage a separate gate-perturbation panel here because the same artefact is read off the residual probe directly; the gate is not a free knob to tune at evaluation time, and its sensitivity to filler distribution is precisely the mechanism the body equations Equations (10) and (11) name.

B.2. Post-training perspective: OLMo-Hybrid cross-stage trajectory

The four-class taxonomy of Section 4.2 is read on the OLMo-Hybrid Base shortlist. As a stability check, we track the same head identities and the same layers whose ablation reduces recall across the three post-training stages of the OLMo-Hybrid release.

Head-level cross-stage trajectory. Figure 9 tracks the top-4 head bundle, ranked by Base mean-ablation ΔR, across Stage 1 and Stage 2 pretraining, Stage 3 long-context training, and the two supervised fine-tuning (SFT) stages, Instruct-SFT and Think-SFT. Ablating the bundle costs more recall than ablating a random size-matched bundle from Stage 3 onward; the gap opens during long-context training, persists through both SFT stages, and the head identities (L11-h$\{1, 9, 13\}$ plus L7-h4) do not reshuffle. The cross-stage view rules out the alternative reading that recall is carried by an ever-shifting set of heads: the model picks a circuit during Stage 3 and keeps it.

Whole-layer cross-stage check. Figure 10 reads the same comparison at the layer granularity: ablating an entire attention layer at the eight indices $\{3, 7, 11, 15, 19, 23, 27, 31\}$ across Base, Instruct-SFT, and Think-SFT. All eight layers contribute

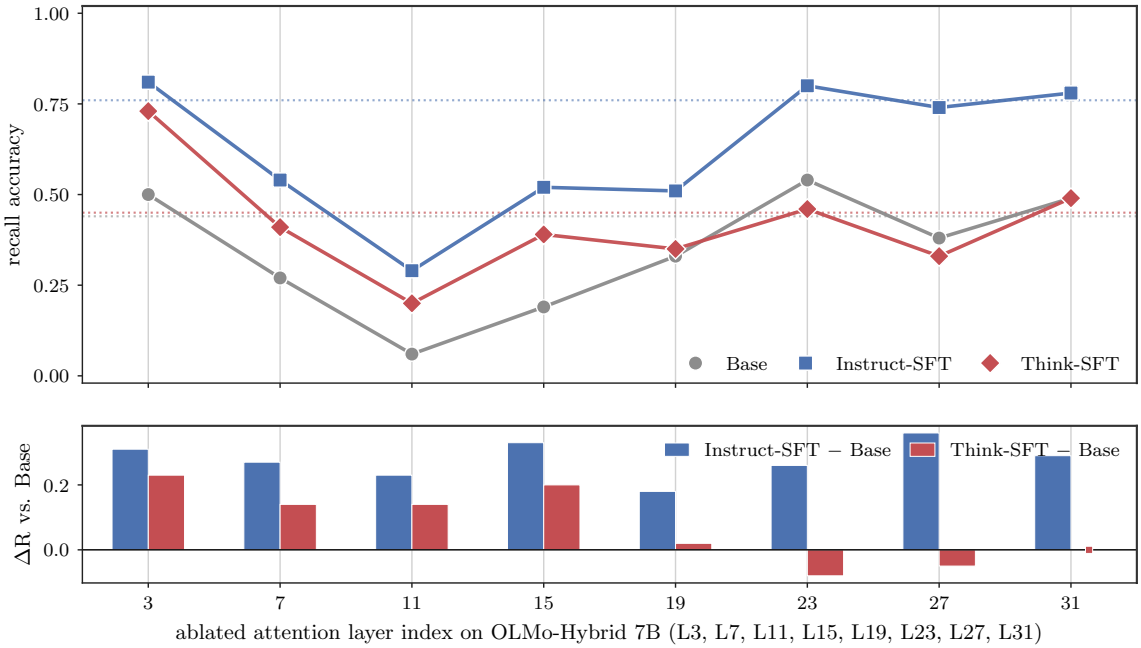

Figure 10. **Whole-layer ablation across post-training: ablating any of the eight OLMo-Hybrid attention layers reduces recall across all three stages, with one notable T-SFT-only signal at L3.** Top: recall accuracy after zero-ablating each attention layer (lines, one per stage), with each stage's un-ablated baseline as a horizontal dotted line (Instruct-SFT ≈ 0.76; Base and Think-SFT ≈ 0.44). Bottom: ΔR relative to the stage's own baseline, showing the dip at L3 under Think-SFT and roughly stable behaviour elsewhere. X-axis is the eight OLMo-Hybrid attention layer indices $\{3, 7, 11, 15, 19, 23, 27, 31\}$. Single ablation seed; K=4, L=512, n=100 prompts. Reads in support of the head-level cross-stage finding (Figure 9): SFT reorganises within rather than across the recall circuit.

to recall in every stage; the only signal that crosses the visible noise band is at L3 under Think-SFT, where ablation costs noticeably more recall than under Base or IS-SFT. The layer-level reading agrees with the head-level one: SFT reorganises within rather than across the existing recall circuit, so the layer-and-head identities tracked at Base are still the right circuit at Think-SFT.

B.3. OLMo-Hybrid four-class taxonomy: shortlist partition and cluster

Table 4 lists the cleanest rows of the four-class taxonomy on OLMo-Hybrid Base under mean ablation in the colors-essay regime, on the 26-head non-silent shortlist (heads with $|\Delta\mathrm{R}^{(h)}_{\mathrm{zero}}| \geq 0.05$). The shortlist partitions as 0 *answer-writer*, 0 *echo-writer*, 3 *structural*, 23 *silent*; the table shows the two cleanest *structural* rows and two of the four *echo-writer* heads excluded from the shortlist (zero-ablation $|\Delta|$ below threshold, hence "–" under ε_Δ=0.05). The full *echo-writer* cluster is L23-h1, L27-h4, L23-h18, L27-h5 at $\mathrm{rate}_{\max 5} \in [0.61, 0.99]$; the first three project the queried colour, the last projects on the broader answer-side template.

Table 4. **Four-class taxonomy on OLMo-Hybrid Base under mean ablation.** Colors-essay regime (K=1, L=512, baseline $recallaccuracy$=0.65); same threshold rule as Table 1.

Class	Head	$\lvert\Delta\mathrm{R}^{(h)}_{\mathrm{mean}}\rvert$	$\mathrm{rate}^{(h)}_{\max 5}$	Top piece
structural	L15-h7	0.110	0.000	`Urb`
structural	L3-h16	0.060	0.000	`oppos`
echo-writer	L23-h1	–	0.990	`blue`
echo-writer	L27-h4	–	0.670	`violet`

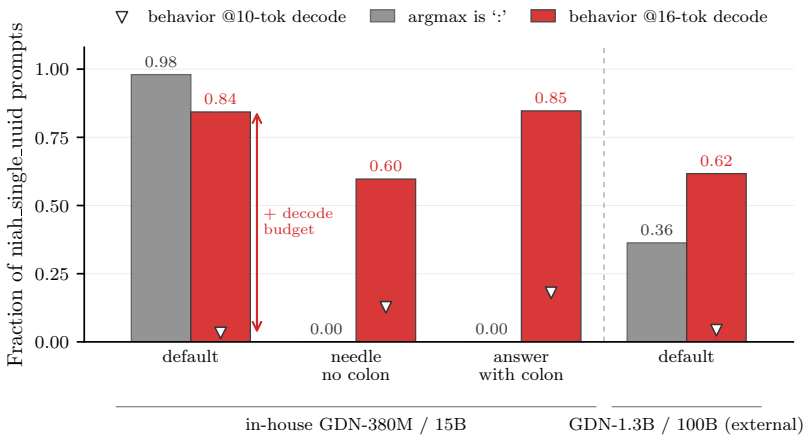

Figure 11. **GDN emits a colon as the first-token argmax on 98% of default-template `niah_single_uuid` prompts; symmetrising the prompt drops this to 0%.** The artefact looks like behavioural failure under a tight decode budget (3.3% behavior at 10 tokens) but the model knows the answer once it is allowed to emit the colon and continue (84.3% at 16 tokens). An independent 1.3B-parameter GDN reproduction trained on 100B tokens (Yang et al., 2025; M-A-P Team, 2024) reads 36.3% in the same direction, so this is template-copying rather than something specific to our checkpoint. We therefore pin the 16-token decode budget across all UUID-style RULER measurements in the paper (Appendix A.5).

B.4. Side-finding: GDN colon-tic, a tokenisation artefact

Reading the per-token argmax of GDN on RULER `niah_single_uuid` reveals a separable failure mode. Under the default RULER template, the in-house GDN checkpoint emits `:` as the first-token argmax on 98% of prompts even though the valid UUID prefix is typically the runner-up. With a 10-token decode budget this collapses behavioural recall to 3.3%; extending the budget to 16 tokens recovers 84.3%, because the model emits the colon and then continues into the UUID. The default `niah_single` needle reads "`... is: <PASSKEY>`" while the canonical answer prefix ends at "`is`", so the colon is a copy from the template into the readout rather than a content failure. Symmetrising the prompt, either by adding the colon to the answer prefix or by removing it from the needle, drives the colon argmax from 0.980 to 0.000 without affecting the valid-prefix rank inside the recurrent state. The artefact is not specific to our checkpoint: an independent 1.3B-parameter GDN reproduction trained on 100B tokens (Yang et al., 2025; M-A-P Team, 2024) reads 36.3% in the same direction. Figure 11 documents this effect; it justifies the 16-token decode budget pinned in Appendix A.5 and shows that linear-attention readouts are particularly exposed to template-copying tics like this one.

C. Related work

Three strands of prior work frame our four contributions: mechanistic studies of natively-pretrained hybrids, recall benchmarks together with retrieval-head papers, and the hybrid-architecture release line. The intro contains the core citations; we expand here on what each strand treats as a default and where we re-open it as a variable. Across the three, attention placement is fixed by a released checkpoint or treated as a uniform-stride knob, the synthetic-versus-text axis is treated as a benchmark choice, and head-level retrieval is read through attention-pattern primitives that capture only a single circuit family. Our placement-controlled sweep, the residual-stream sequence-layer map, the head-level four-class taxonomy with the *echo-writer* class, and the filler-regime stress test address each of those defaults in turn.

Mechanistic work on natively-pretrained hybrids. The closest prior art opens natively-pretrained hybrids one component at a time and reports where retrieval lives. Bick et al. (2025) identify a gather-and-aggregate circuit at specific layers of recurrent and hybrid models, and Bick et al. (2026) apply the same analysis to retrieval-aware distillation, both treating attention placement as fixed by the released checkpoint. Michalak & Abreu (2025) ablate attention versus linear-recurrent blocks across several hybrid releases and report H1, that linear-attention layers contribute zero retrieval signal at the PREDICT position; Arora et al. (2025) run a parallel evaluation contrasting transformers with state-space models on synthetic and natural data, and Borobia et al. (2026) catalog functional specialisation patterns across architecture families through component ablation. A second body of work imports transformer-circuit methodology into pure linear-attention and state-space models without crossing into hybrids: linear probing on intermediate states (Alain & Bengio, 2016; Hewitt & Liang, 2019) is the workhorse we reuse for the sequence-layer map; logit-lens style decoding and contrastive activation

reading are applied to Mamba and similar models in Ali et al. (2024); Paulo et al. (2025), activation editing based on Rank-One Model Editing and indirect-object-identification circuit replication are ported in Sharma et al. (2024); Ensign & Garriga-Alonso (2024); Endy et al. (2025), and Wang et al. (2024); Airlangga et al. (2025); Merrill et al. (2024) characterise dictionary universality and the recurrent state-tracking limits these methods land in. Our placement-controlled training sweep, the sequence-layer map as a sequence-and-layer diagnostic, and the head-level four-class taxonomy together take a different approach: instead of opening one released checkpoint, we vary attention placement at pretraining time and read the residual stream at every position-layer pair. The sweep makes the architectural variable explicit rather than absorbing it into the choice of release. Our reading is compatible with their H1 rather than against it. In OLMo-Hybrid we observe probe accuracy near 1.0 on the residual at depth $\ell{=}1$ at the POST-NEEDLE anchor, but the probe reads decodability of the residual at depth $\ell{=}1$ and does not isolate the L1 linear-attention mixer's contribution from the embedding and the L0 write that share the same residual; the reconciliation with H1 is therefore a granularity contrast between behaviour at PREDICT (which H1 measures) and decodability at POST-NEEDLE (which our probe measures), and a causal companion that pairs the probe with an L1 attention-output ablation, flagged in Section 5, is needed to attribute the encoding to the mixer itself.

Recall benchmarks, retrieval heads, and the synthetic-versus-text axis. Prior work closest to our methodology measures recall behaviourally and focuses largely on transformers. Arora et al. (2024a) introduce multi-query associative recall as a recall stress test on a controlled synthetic distribution and show that this single deficit drives most of the perplexity gap between linear-attention and quadratic-attention models; Arora et al. (2024b) build the linear-attention design line that the synthetic isolates. Hsieh et al. (2024) build RULER as the long-context benchmark we adopt, with Kuratov et al. (2024) arguing for natural-text haystacks on distributional-plausibility grounds and Jelassi et al. (2024) establishing that recurrent architectures can copy via verbatim repetition without paying the recall cost; Okpekpe & Orvieto (2025) revisit associative recall in modern recurrent models. At the head level, Wu et al. (2025) identify retrieval heads via attention-mass-on-needle in dense transformers; Singh et al. (2024) and Olsson et al. (2022) catalog the induction-head precursors of this circuit. The shared scope across this line is behavioural accuracy or transformer-only attention-head attribution, with the synthetic-versus-text choice and the filler regime treated as evaluation defaults rather than independent variables. We re-open both axes: the synthetic-versus-text axis is run as a contrast rather than a selection, and the filler regime is treated as a stress test on the recurrent state, with the rank-one cancellation in the gated delta rule (Section 4.3) giving a mechanistic reason why a haystack of repeated keys is a no-op on linear-attention memory while a haystack of distinct keys is not. Our head-level analysis of natively-pretrained hybrids extends the retrieval-head primitive of Wu et al. (2025) with a projection axis, and distinguishes an *echo-writer* class that their attention-pattern primitive misses.

Hybrid architectures and design choices. The hybrid architecture line is dense and largely design-and-evaluation rather than mechanism. Lieber et al. (2024); MiniMax Team (2025); Ren et al. (2024); Botev et al. (2024); De et al. (2024); Glorioso et al. (2024b;a) introduce mixed transformer and linear-attention stacks at varying ratios, using downstream benchmark numbers as the primary evidence; Dong et al. (2025) place attention and recurrence in parallel head groups rather than in series, while Zuo et al. (2025); Qwen Team (2025); NVIDIA (2025); IBM Research (2025) ship industrial hybrid families with their own placement and ratio choices, and Merrill et al. (2026) accompany the OLMo-Hybrid release with theory-meets-practice ablations. The most systematic academic studies sweep the attention-to-recurrence ratio across dozens of models while holding stride uniform (Wang et al., 2025a; Waleffe et al., 2024), and theory shows that a single quadratic-attention layer suffices for in-context retrieval without specifying its depth (Wen et al., 2025). Cross-architecture distillation studies have already noted that RULER tracks the number of preserved quadratic-attention layers while a nine-task LM-eval average stays flat across the same sweep (Li et al., 2025), and the quadratic-attention controls we reuse here for design-side context (Vaswani et al., 2017; Ainslie et al., 2023; DeepSeek-AI, 2024) fix the dense baselines our placement sweep is read against. Across this set, attention placement is a design knob fixed before training and reported as a recipe rather than studied as an independent variable, and circuit-level analysis is either absent or limited to one released checkpoint at a time. Our placement-controlled sweep over single-attention positions, plus linear-attention and quadratic-attention controls, treats placement as the variable to be probed, which is the gap our intro contribution (a) addresses.

D. Per-seed disaggregation of the body figures

Figures 4 and 5 aggregate over training seeds. Two patterns the aggregation hides are visible per-seed and worth a closer look. Single-attention placements at L4 and L8 are bimodal: one training seed concentrates probe accuracy at the prediction position while another remains near zero, so the across-seed mean misrepresents both modes. A small number of

(architecture, filler) eval cells render as gray "no run" tiles rather than blanks; the trained checkpoints span the full $\{1, 2, 42\}$ seed set for every architecture, the gap is in the eval JSONs only. Figures 12 and 13 show the primary *essay* filler regime; the *repeat* filler foil is omitted because per-seed dispersion compresses on that regime (the recurrent state is not contested) and the panel adds no information beyond the body figure.

D.1. Per-task LM-eval numbers

The 9-task LM-eval average plotted in Figure 1 and protocol-pinned in Appendix A.7 is broken out per task and per checkpoint in Table 5, and per task and per training seed in Table 6. Values are zero-shot `acc,none` from the EleutherAI `lm-evaluation-harness` (Gao et al., 2024); the rightmost column is the unweighted mean of the nine task columns.

Table 5. **Per-task zero-shot LM-eval accuracy at 340M parameters, 15B SlimPajama tokens.** From the EleutherAI `lm-evaluation-harness` (Gao et al., 2024). Cells are the median across training seeds (column "seeds"). The rightmost column averages the nine task columns and provides the value plotted in Figure 1b. The body refers to this whole-table spread of 0.423 to 0.451, a range of 0.028, as "LM-eval insensitivity to placement."

Architecture	seeds	arc-c	arc-e	boolq	hella.	lambada	openb.	piqa	sciq	winog.	9-task avg
single-attn L0	$\{1, 2, 42\}$	0.199	0.480	0.583	0.315	0.333	0.184	0.656	0.792	0.516	**0.451**
single-attn L4	$\{1, 2, 42\}$	0.189	0.465	0.566	0.311	0.317	0.192	0.653	0.784	0.507	0.443
single-attn L8	$\{1, 2, 42\}$	0.197	0.464	0.553	0.314	0.318	0.180	0.650	0.796	0.510	0.442
single-attn L12	$\{1, 2, 42\}$	0.196	0.470	0.576	0.314	0.321	0.180	0.653	0.782	0.504	0.444
single-attn L16	$\{1, 2, 42\}$	0.193	0.468	0.569	0.314	0.337	0.176	0.645	0.792	0.507	0.445
single-attn L19	$\{1, 2, 42\}$	0.195	0.461	0.567	0.313	0.354	0.170	0.652	0.775	0.504	0.443
MHA	$\{1, 2, 42\}$	0.189	0.470	0.587	0.314	0.343	0.180	0.649	0.802	0.523	**0.451**
GQA	$\{2, 42\}$	0.190	0.469	0.606	0.314	0.329	0.182	0.653	0.760	0.515	0.447
MLA	$\{1, 2\}$	0.188	0.423	0.612	0.288	0.275	0.154	0.621	0.730	0.525	0.424
GDN	$\{1, 2, 42\}$	0.193	0.451	0.575	0.302	0.289	0.176	0.640	0.784	0.500	0.434
GLA	$\{2, 42\}$	0.185	0.433	0.568	0.294	0.269	0.163	0.631	0.748	0.516	0.423

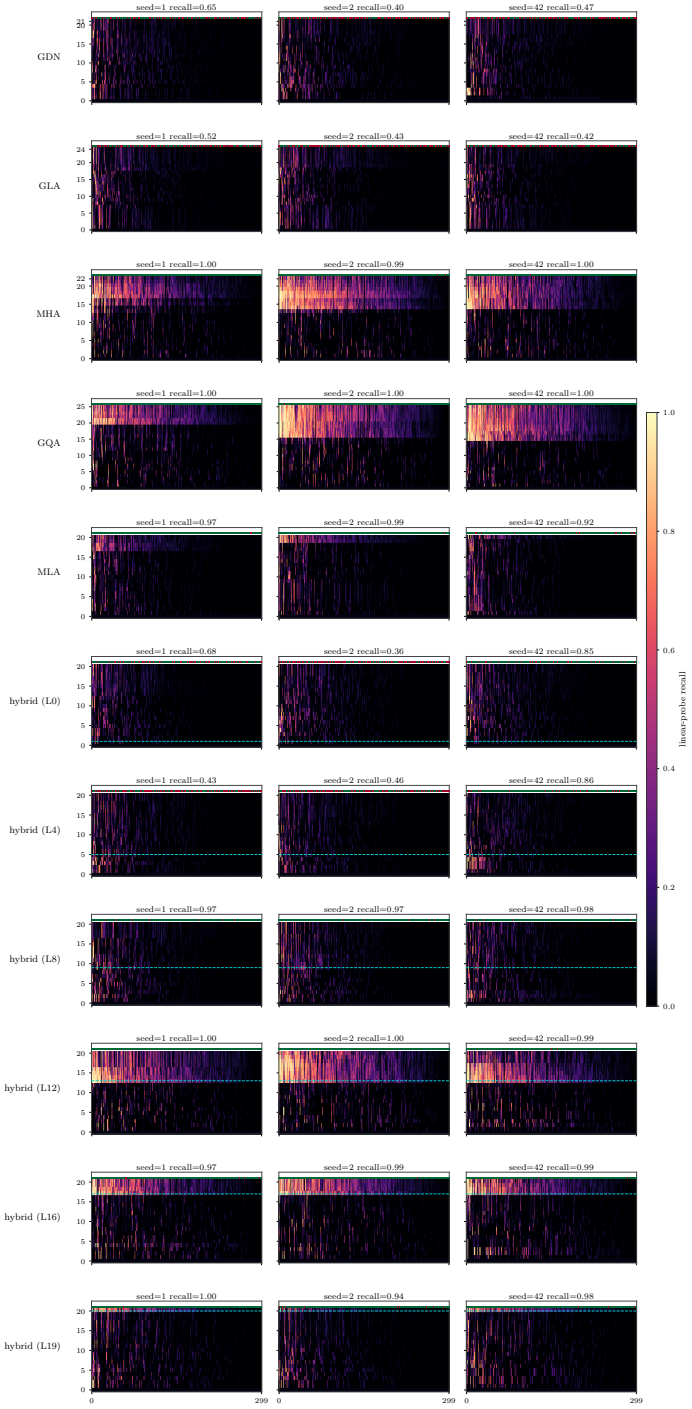

Figure 12. **Per-seed disaggregation of Figure 4 for `niah_single_num` under *essay* filler.** Rows are corpus models in canonical order; columns are training seeds $\{1, 2, 42\}$. Each cell is a layer $\times$ doc-sorted-by-peak heatmap (mean over five probe seeds for that one training seed). Cyan dashed line marks the attention-layer row in single-attention hybrids; quadratic-attention rows have no marker. The L4 and L8 single-attention placements split per-seed: one seed concentrates near the prediction position while another remains near zero, which the across-seed mean in Figure 4 smooths out. Cells labelled "no run" are eval JSONs not yet collected; the corresponding checkpoints exist.

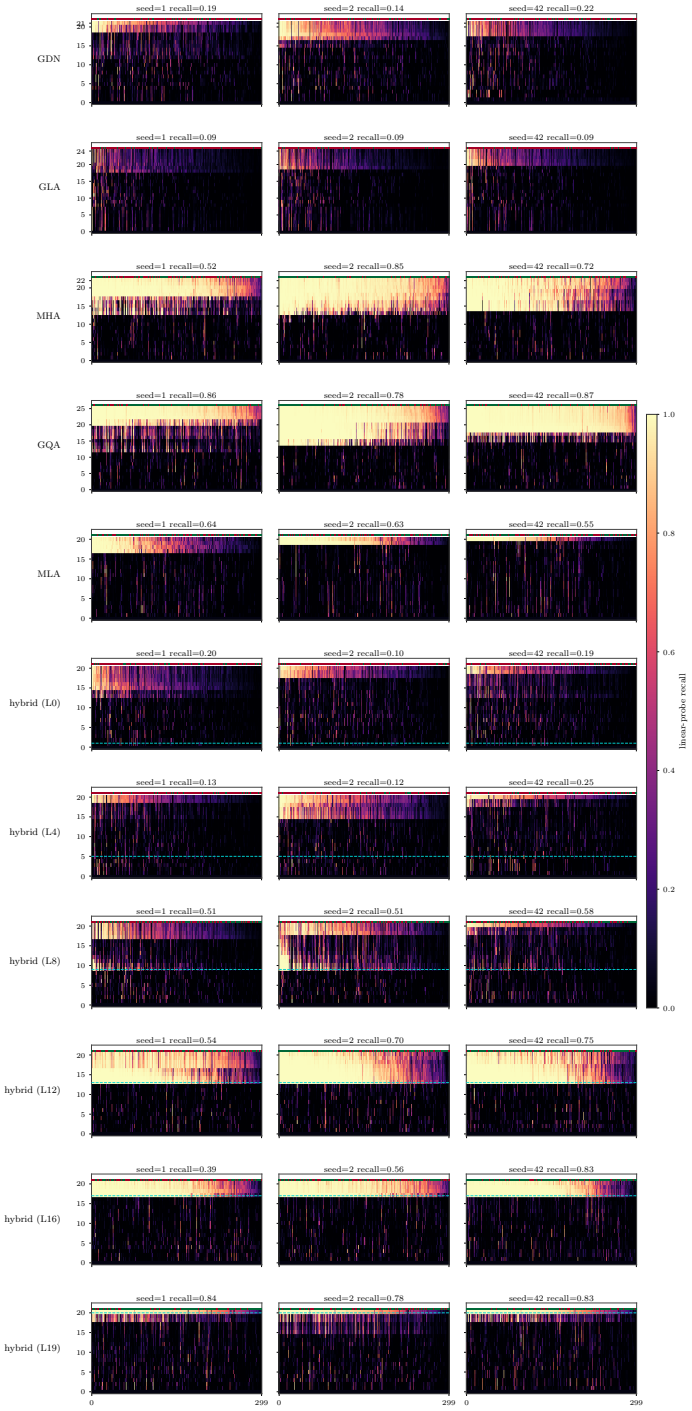

Figure 13. **Per-seed disaggregation of Figure 4 for `niah_single_uuid` under *essay* filler.** Same axes as Figure 12. UUID is the more discriminating task: late-placement single-attention placements and the quadratic-attention controls (MHA, GQA) concentrate at the prediction position; early-placement single-attention and the linear-attention controls (GDN, GLA) stay dark across all seeds.

Table 6. **Per-seed breakdown of Table 5.** Cell values are zero-shot `acc, none` from the EleutherAI `lm-evaluation-harness` (Gao et al., 2024) for the listed training seed. Across-seed spread on the 9-task average is small relative to the architecture spread, so the medians in Table 5 are representative.

Architecture	seed	arc-c	arc-e	boolq	hella.	lambada	openb.	piqa	sciq	winog.	9-task avg
single-attn L0	1	0.199	0.480	0.583	0.315	0.326	0.186	0.656	0.792	0.510	0.450
single-attn L0	2	0.191	0.473	0.517	0.313	0.333	0.180	0.651	0.786	0.533	0.442
single-attn L0	42	0.202	0.489	0.589	0.333	0.371	0.184	0.671	0.818	0.516	0.464
single-attn L4	1	0.182	0.464	0.574	0.311	0.309	0.192	0.653	0.775	0.507	0.441
single-attn L4	2	0.195	0.465	0.558	0.311	0.320	0.192	0.657	0.784	0.496	0.442
single-attn L4	42	0.189	0.475	0.566	0.314	0.317	0.182	0.653	0.807	0.512	0.446
single-attn L8	1	0.197	0.464	0.548	0.313	0.317	0.180	0.644	0.791	0.496	0.439
single-attn L8	2	0.199	0.485	0.553	0.314	0.318	0.186	0.650	0.809	0.522	0.448
single-attn L8	42	0.187	0.462	0.573	0.314	0.327	0.160	0.655	0.796	0.510	0.443
single-attn L12	1	0.191	0.457	0.576	0.314	0.321	0.182	0.643	0.782	0.504	0.441
single-attn L12	2	0.196	0.470	0.599	0.314	0.321	0.164	0.653	0.780	0.504	0.445
single-attn L12	42	0.200	0.476	0.535	0.318	0.332	0.180	0.665	0.782	0.499	0.443
single-attn L16	1	0.182	0.464	0.569	0.314	0.344	0.176	0.658	0.792	0.507	0.445
single-attn L16	2	0.201	0.468	0.596	0.312	0.335	0.176	0.643	0.793	0.514	0.449
single-attn L16	42	0.193	0.469	0.549	0.315	0.337	0.178	0.645	0.786	0.497	0.441
single-attn L19	1	0.188	0.468	0.567	0.313	0.358	0.170	0.653	0.798	0.527	0.449
single-attn L19	2	0.198	0.458	0.560	0.313	0.329	0.196	0.644	0.775	0.494	0.441
single-attn L19	42	0.195	0.461	0.596	0.313	0.354	0.166	0.652	0.772	0.504	0.446
MHA	1	0.196	0.470	0.602	0.314	0.343	0.180	0.656	0.799	0.528	0.454
MHA	2	0.189	0.488	0.541	0.314	0.358	0.186	0.646	0.808	0.497	0.447
MHA	42	0.189	0.469	0.587	0.317	0.341	0.162	0.649	0.802	0.523	0.449
GQA	1	0.191	0.462	0.588	0.314	0.318	0.180	0.650	0.795	0.511	0.446
GQA	2	0.195	0.468	0.606	0.315	0.334	0.176	0.652	0.764	0.521	0.448
GQA	42	0.185	0.471	0.605	0.314	0.324	0.188	0.654	0.757	0.510	0.445
MLA	1	0.184	0.425	0.604	0.288	0.273	0.148	0.627	0.731	0.511	0.421
MLA	2	0.191	0.422	0.621	0.288	0.278	0.160	0.616	0.729	0.539	0.427
MLA	42	0.176	0.424	0.614	0.293	0.268	0.158	0.636	0.752	0.539	0.429
GDN	1	0.179	0.437	0.575	0.302	0.276	0.174	0.640	0.784	0.489	0.428
GDN	2	0.193	0.451	0.550	0.297	0.289	0.182	0.634	0.759	0.500	0.428
GDN	42	0.196	0.462	0.578	0.311	0.322	0.176	0.642	0.787	0.506	0.442
GLA	1	0.178	0.427	0.594	0.295	0.266	0.162	0.624	0.738	0.522	0.423
GLA	2	0.179	0.427	0.598	0.290	0.251	0.172	0.623	0.755	0.514	0.423
GLA	42	0.191	0.438	0.539	0.298	0.287	0.154	0.639	0.742	0.519	0.423