

What Does Layer-Importance Reveal About Transformers and State-Space Models?

Istabrak Abbes^{1,2,3}, Nizar Islah^{2,3}, Irina Rish^{2,3,5}, Sarath Chandar^{1,2,4,5}

¹Chandar Research Lab, ²Mila - Quebec AI Institute, ³Université de Montréal

⁴Polytechnique Montréal, ⁵Canada CIFAR AI Chair

istabrak.abbes@mila.quebec

Abstract

Transformers and state-space models (SSMs) are the two dominant families of sequence models, and a central open question is how far the analytical knowledge built for transformers transfers to SSMs. We address this through the lens of layer importance which underpins compression, selective fine-tuning, and interpretability across both families. We decompose layer importance into two distinct notions. *Necessity* captures how much the pretrained model depends on a layer’s existing contribution, measured by the loss increase from bypassing it. *Plasticity* captures where the model absorbs new information during fine-tuning, measured by the magnitude of task-specific weight updates. Our analysis reveals that the two families behave fundamentally differently: in every evaluated residual transformer up to 14B parameters, Necessity and Plasticity anti-align across depth, whereas in the evaluated Mamba-style SSMs they point to overlapping regions. The sign of this alignment also predicts downstream adaptation behavior. In the evaluated transformers, concentrating updates in the most plastic layers increases catastrophic forgetting, while this tier-dependent effect disappears in the evaluated Mamba-style SSMs.

1 Introduction

Modern language modeling is still dominated by residual transformers (Vaswani et al., 2023), but selective state-space models have emerged as a major alternative architecture (Gu and Dao, 2024; Dao and Gu, 2024). A large body of work on transformers has produced mature tools for pruning (Gromov et al., 2025; Men et al., 2025), parameter-efficient fine-tuning (Hu et al., 2021), and interpretability (Meng et al., 2023; Song et al., 2026), many of which rely on estimating *layer importance* (Zhang et al., 2024; Naim et al., 2026; Xu et al., 2026; Yao et al., 2024). On the other hand, SSMs inherit the same vocabulary but not, obviously, the same internal structure. The natural question is whether the

analytical tools calibrated on transformers transfer to SSMs, or whether they silently measure something else when selective recurrent state dynamics replace attention-based token mixing. We study this question by decomposing layer importance into two distinct notions. *Necessity* measures how much the pretrained model depends on a layer’s existing contribution, quantified through the loss increase caused by bypassing the layer. *Plasticity* measures where the model absorbs new information during adaptation, quantified through the magnitude of task-specific weight updates during fine-tuning. Although these notions are often conflated under a single notion of “importance,” they capture fundamentally different properties: pruning and interpretability primarily rely on Necessity (Men et al., 2025; Zhang et al., 2024), while selective fine-tuning relies on Plasticity (Pan et al., 2024; Zhang et al., 2023; Qing et al., 2024; Hayou et al., 2026). The key question is whether these two quantities identify the same regions of a network, and whether this relationship transfers across architectures.

To make this distinction explicit, we introduce *Necessity-Plasticity Alignment* ($\mathcal{NPA}$): the mean Spearman correlation, across tasks, between an ablation-based Necessity ranking and a LoRA-update-magnitude-based Plasticity ranking of the model’s layers. Positive $\mathcal{NPA}$ means that the same layers are both necessary and plastic; negative $\mathcal{NPA}$ means that pretrained dependence and adaptation concentrate at opposite ends of the network; values near zero indicate that no stable depth ordering exists without specifying the estimator.

Across the evaluated transformer and Mamba-style checkpoints, $\mathcal{NPA}$ separates the two families cleanly. Every evaluated residual transformer up to 14B parameters exhibits negative alignment: early layers are most necessary, while late layers are most plastic. In contrast, every evaluated Mamba-style SSM exhibits positive alignment, with Ne-

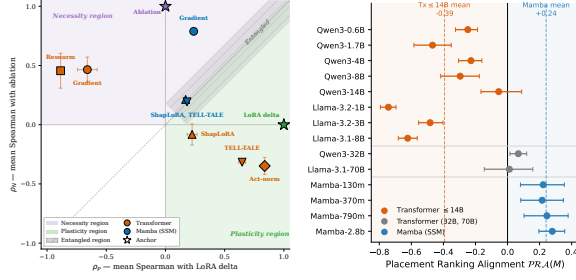


Figure 1: **Layer importance is method-relative and architecture-conditional.** **Left:** Estimators projected onto the Necessity–Plasticity plane (mean Spearman correlation with ablation, vertical; with LoRA-delta, horizontal). In transformers, gradient and resnorm methods align with Necessity; activation norm and TELL-TALE align with Plasticity. In Mamba, estimators cluster in the mixed-positive region. **Right:** $\mathcal{NPA}(M)$ across checkpoints (bars: bootstrap 95% CI). Every evaluated transformer up to 14B is negative; every evaluated Mamba-style SSM is positive; the two largest transformers collapse toward zero. Dashed lines mark family means.

cessity and Plasticity concentrated in overlapping regions. This contrast persists across importance estimators, survives removal of the first layer, and extends beyond Mamba to RWKV and a hybrid architecture. It is also not an artifact of low-rank adaptation: an adapter-free control that recomputes Plasticity from full-parameter fine-tuning preserves the sign of $\mathcal{NPA}$ on every checkpoint we retrained (Appendix E.3). Larger transformers, including Qwen3-32B and Llama-3.1-70B, do not enter the SSM regime; instead, their alignment collapses toward zero. The same split predicts downstream adaptation behavior: transformers show tier-dependent forgetting when successive tasks adapt highly plastic layers, whereas this effect vanishes in Mamba-style SSMs.¹

Contributions. 1) We introduce *Necessity-Plasticity Alignment* ($\mathcal{NPA}$), a diagnostic that measures the agreement between pretrained layer dependence and task-specific adaptation across depth.

2) We find a robust architecture-dependent split: every evaluated residual transformer up to 14B exhibits negative $\mathcal{NPA}$, with early layers most necessary and late layers most plastic, whereas the evaluated Mamba-style SSMs exhibit positive $\mathcal{NPA}$, with Necessity and Plasticity concentrated in over-

lapping regions. The split survives an adapter-free full fine-tuning control and reappears *within* a single hybrid checkpoint.

3) We show that $\mathcal{NPA}$ predicts selective fine-tuning behavior in a controlled two-task continual-learning probe: the evaluated transformers exhibit strong forgetting when both tasks adapt the same highly plastic layers, while this tier-dependent interference disappears in the evaluated Mamba-style SSMs.

2 Related Work

Importance-guided PEFT and rank/placement allocation. A growing line of parameter-efficient fine-tuning (PEFT) work argues that adaptation should not be uniform across depth. AdaLoRA (Zhang et al., 2023) reallocates rank by singular-value importance during training; AlphaLoRA (Qing et al., 2024) assigns LoRA experts using heavy-tailed self-regularization statistics; Layer Card (Xu et al., 2026) introduces *resnorm* as a reusable layer diagnostic for selective placement; ShapLoRA (Zhao et al., 2026) uses Shapley-style sensitivity scores; and PLoP (Hayou et al., 2026) proposes a precise placement signal driven by neural-feature norms. Each of these methods proposes a single importance estimator and studies it in isolation. Recent work has also begun to ask whether compression, PEFT, and interpretability tools developed for transformers transfer to selective state-space models. LoRA-style adaptation has been extended to Mamba-style architectures (Galim et al., 2025), while layer pruning and relevance-propagation methods have been adapted to selective SSMs (Munoz et al., 2025; Jafari et al., 2025). These works motivate cross-architecture comparisons, but they do not ask whether different task-sensitive importance estimators agree on the same checkpoints

Layer roles, redundancy, and depth-dependent task usage. A parallel literature studies which layers are useful, redundant, or task-specific in pretrained models. ShortGPT (Men et al., 2025) shows that many transformer layers can be removed with limited degradation; TELL-TALE (Naim et al., 2026) performs task-aware layer elimination; Song et al. (2026) argue that depth contributes differently to retrieval, knowledge, and reasoning; Yao et al. (2024) study layer-wise importance for memory-efficient PEFT. Zhang et al. (2024) identify “cornerstone” layers via Shapley-style ablation, finding

¹Code, per-layer importance scores, and analysis scripts: <https://github.com/chandar-lab/layer-importance-ssm-vs-transformers>.

early-layer dominance, a phenomenon we recover under ablation but show dissolves under LoRA delta on the same checkpoint. Nepal et al. (2025) report that the layers ablation identifies as important for mathematical reasoning are stable across pretraining and post-training. Shi et al. (2025) learn binary masks for layer significance during alignment and observe high overlap across alignment datasets, all under a single estimator. These results document stability *within* one estimator; we document the orthogonal fact that *across* estimators, the very identity of “important” layers changes, and that whether it changes is architecture-conditional.

Method disagreement in attribution and importance. Disagreement between attribution methods is an established finding in the explainability literature. Adebayo et al. (2020) show that gradient-saliency maps survive parameter randomizations that destroy network behavior, evidence that gradient attribution does not measure the same thing as causal intervention. Ancona et al. (2018) unify gradient-based methods as varying-fidelity linearizations of true ablation; Sundararajan et al. (2017) prove that no attribution method satisfies sensitivity, implementation invariance, and completeness simultaneously. Krishna et al. (2024) measure the magnitude of feature-level attribution disagreement and characterize how practitioner interpretations shift with the method chosen. Our contribution shifts the disagreement from feature attributions to layer rankings, from a single architecture to a transformer/SSM contrast, and from “expected at high resolution” to a structurally predictable function of residual-stream gradient flow.

3 Layer-Importance Diagnostic Protocol

Our goal is to determine whether commonly used layer-importance estimators measure the same underlying property across architectures, or whether their behavior depends on the distinction between pretrained reliance and adaptation dynamics. We therefore study layer importance through two complementary quantities: *Necessity* and *Plasticity*.

Necessity. Necessity measures how much the pretrained model depends on a layer’s existing contribution at inference time. A layer is considered necessary if bypassing it substantially degrades performance. For a model M with layers $\{\ell_i\}_{i=1}^L$, we define the Necessity score of layer ℓ_i on task t as:

$$N(\ell_i; M, t) = \mathcal{L}(M_{\setminus \ell_i}, t) - \mathcal{L}(M, t),$$

where $\mathcal{L}$ denotes the task loss and $M_{\setminus \ell_i}$ is the model with layer ℓ_i bypassed. Larger values indicate that the pretrained computation relies more strongly on that layer.

Plasticity. Plasticity measures where the model absorbs new information during adaptation. Rather than quantifying reliance on the pretrained computation, it measures how strongly each layer changes during task-specific fine-tuning. Using LoRA adaptation, we define the Plasticity score of layer ℓ_i as:

$$P(\ell_i; M, t) = \left\| \Delta W_i^{(t)} \right\|_F,$$

where $\Delta W_i^{(t)}$ is the learned LoRA update for layer ℓ_i on task t , and $\|\cdot\|_F$ denotes the Frobenius norm. Larger values indicate that adaptation concentrates more strongly in that layer.

Although both quantities are often grouped under a single notion of “importance,” they capture fundamentally different properties. Necessity measures dependence of the pretrained model on an existing computation, whereas Plasticity measures receptivity to new information during adaptation. The relationship between these two quantities is therefore an empirical and architectural question.

To study this relationship, we introduce *Necessity-Plasticity Alignment* ($\mathcal{NPA}$), defined as the Spearman correlation between layer rankings induced by Necessity and Plasticity:

$$\mathcal{NPA}(M, t) = \rho_{\text{Spearman}}(R_{\text{Nec}}(M, t), R_{\text{Plast}}(M, t)),$$

where R_{Nec} and R_{Plast} denote the corresponding layer rankings for model M on task t .

Positive $\mathcal{NPA}$ indicates that the same layers are both necessary and plastic, while negative $\mathcal{NPA}$ indicates that pretrained reliance and adaptation concentrate on opposite ends of the network. Values near zero indicate that no stable agreement exists between the two notions of importance.

This protocol allows us to compare layer-importance structure across architectures independently of any single estimator. Rather than asking whether one importance metric is universally correct, we ask whether different estimators consistently align with Necessity or Plasticity, and whether the relationship between these quantities changes across model families.

4 Transformers and SSMs Disagree on Layer Importance

We now apply the diagnostic protocol to the transfer question directly. If transformer-derived layer-importance concepts transferred uniformly to SSMs, Necessity and Plasticity rankings should relate similarly across both families. Instead, their relationship changes sign across architectures.

4.1 Necessity–Plasticity Alignment changes sign across architectures

Figure 1 reports $\mathcal{NPA}(M)$ between ablation-based Necessity rankings and LoRA-delta-based Plasticity rankings. The sign of $\mathcal{NPA}$ cleanly separates the two families. All evaluated Mamba-style SSMs show positive alignment: layers receiving larger LoRA updates are also those whose ablation causes larger loss increases. In contrast, every evaluated residual transformer up to 14B shows negative alignment: the layers most necessary for preserving pretrained computation are not the layers where fine-tuning writes the largest task-specific updates. Figure 2 resolves the same measurement per (model, task) pair and shows that the split is not carried by a handful of outlier tasks: within each family the sign is consistent across the task suite.

The largest transformers weaken this pattern without becoming SSM-like. Qwen3-32B and Llama-3.1-70B move toward near-zero alignment, suggesting a partial reorganization of Plasticity rather than recovery of a universal layer ordering. This disagreement is not estimator noise: LoRA-delta rankings remain self-consistent under matched scopes while continuing to disagree with ablation. Strong negative $\mathcal{NPA}$ requires a concentrated high-Necessity region together with an adaptation profile that avoids it, and both ingredients weaken with scale; since $\mathcal{NPA}$ is a rank correlation, flattening either profile drives it toward zero. We therefore state the negative-transformer result for the evaluated $\leq 14B$ regime and do not extrapolate beyond the measured checkpoints.

The split is not an artifact of low-rank adaptation. Because Plasticity is read off a LoRA update, the split could in principle reflect where LoRA places updates rather than a property of the architecture. An adapter-free control on 13 checkpoints rules this out (Appendix E.3): full-FT and LoRA update profiles are positively rank-correlated on every one, and the sign of the alignment is preserved throughout, negative for every evaluated

transformer with ablation baselines and positive or near zero for the evaluated Mamba, RWKV, and hybrid checkpoints. Normalizing updates by pre-trained weight norm leaves this unchanged. The conclusion therefore does not depend on LoRA rank, scaling, initialization, or target-module selection.

Generalization beyond Mamba. The positive-alignment regime is not specific to Mamba. RWKV (Peng et al., 2023) exhibits strongly positive alignment ($\mathcal{NPA} = +0.54$, with all per-task correlations positive). Zamba2 (Glorioso et al., 2024), a hybrid SSM+attention model, lies near zero but slightly positive ($\mathcal{NPA} = +0.05$), while Falcon-Mamba-7B (Zuo et al., 2024) lies slightly below zero ($\mathcal{NPA} = -0.02$). Together, these results suggest the ordering

$$\mathcal{NPA}_{\text{SSM}} > \mathcal{NPA}_{\text{Hybr.}} \approx 0 > \mathcal{NPA}_{\text{Transf.} \leq 14B}.$$

The transformer-SSM contrast is strongest at small and medium scale, while very large transformers and boundary SSMs move toward a shared near-zero regime.

Necessity and Plasticity also dissociate within a hybrid model. Zamba2-2.7B interleaves 9 shared-attention blocks among 45 Mamba blocks, which lets us test the dissociation without changing checkpoints. Attention blocks are only 16.7% of the layers but absorb 36.3% of LoRA and 31.6% of full-FT update mass, while carrying just 10.4% of ablation Necessity mass, which concentrates in the Mamba blocks (Appendix E.4). Zamba2’s near-zero whole-model $\mathcal{NPA}$ is therefore the average of a transformer-like and an SSM-like component, not the absence of the effect.

4.2 Depth profiles explain the architecture split

Figure 3 shows the spatial origin of the sign change. In transformers, Plasticity concentrates near the output side of the network, whereas Necessity concentrates near the input side, often dominated by a strong layer-0 component. The two quantities therefore select opposite ends of depth, producing negative alignment. In Mamba-style SSMs, the profiles are different but not spatially opposed. Plasticity peaks in middle layers, while Necessity emphasizes boundary layers with substantial overlap. Because the two quantities co-occupy partially overlapping regions, the resulting alignment remains positive.

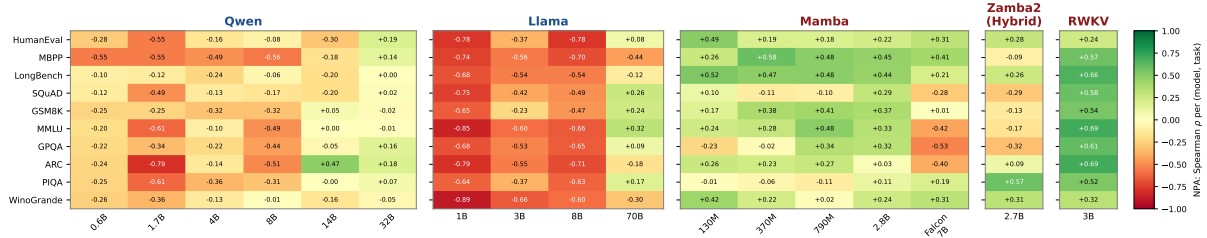


Figure 2: **Per-task $\mathcal{NPA}$ across architectures and scales.** Spearman ρ between Necessity and Plasticity rankings per (model, task) pair. Transformers are predominantly negative (red) up to 14B, fading toward zero at larger scale. Mamba-family and RWKV6-3B are positive (green). Hybrid and boundary models sit near zero.

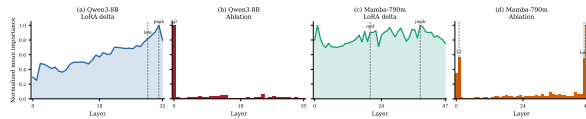


Figure 3: **Representative Necessity and Plasticity depth profiles.** Qwen3-8B: LoRA delta concentrates near the final layers (a); ablation is dominated by an early layer-0 spike (b). Mamba-790m: LoRA delta peaks mid-network (c); ablation emphasizes boundary layers with partial overlap (d). Transformers select opposite ends of depth; Mamba profiles differ but are not spatially opposed.

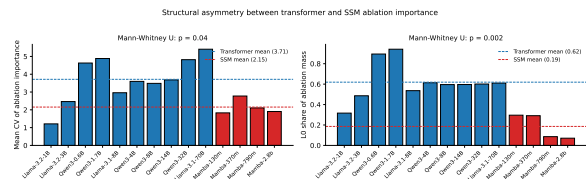


Figure 4: **Leave-out-L0 and L0-share controls per checkpoint.** Transformers concentrate more ablation mass on layer 0 than Mamba models, yet removing layer 0 shifts transformer alignment toward zero without reversing its sign, ruling out the embedding bottleneck as the sole driver of the architecture split.

This depth organization also explains why large transformers move toward zero alignment. Their Necessity profiles remain strongly early-layer dominated, but Plasticity becomes less purely terminal and develops additional early-layer mass. The resulting partial overlap weakens anti-alignment without producing the positive agreement observed in SSMs.

4.3 Layer-0 and metric controls

A natural alternative explanation is that the transformer–SSM contrast is driven entirely by layer 0. Figure 4 rules out this strong form. Removing layer 0 shifts transformer checkpoints toward zero, but every evaluated transformer up to 14B

remains negatively aligned, while Mamba checkpoints change minimally. Transformers do allocate substantially more Necessity mass to layer 0 than Mamba models, but the anti-alignment persists even after removing it, indicating a broader depth-wise separation between Necessity and Plasticity.

We also test whether ranking-based placement conclusions transfer uniformly across validation metrics. They do not. On multiple-choice evaluation, the relative advantage of top- k versus bottom- k placement depends on both the estimator and the model family. Even after rankings are computed under a shared proxy protocol, their downstream interpretation remains metric-relative.

4.4 Estimator spectrum and implications

The Necessity–Plasticity distinction also organizes estimators beyond the two anchors. In transformers, gradient- and resnorm-based estimators align more strongly with Necessity, while activation norm and TELL-TALE align more strongly with Plasticity. ShapLoRA occupies an intermediate position. In Mamba, where Necessity and Plasticity are already positively aligned, the estimators cluster within the same mixed-positive region.

These results show that layer importance is not a universal ordering of depth. Ablation-based, adaptation-based, and proxy estimators are not noisy measurements of a single latent ranking; they measure different properties of the model. A layer can be important because removing it disrupts pre-trained computation, or because fine-tuning preferentially writes into it.

Importance-guided placement should therefore be treated as a measurement problem rather than a model-independent fact.

Forgetting at comparable scale (~3B): SSMs vs Transformers

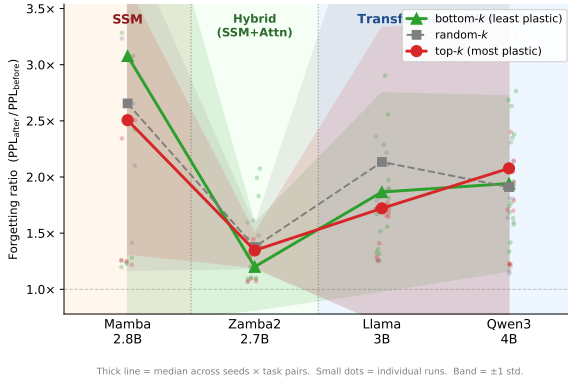


Figure 5: **Tier-dependent forgetting is a transformer effect.** Forgetting ratio ($PPL_{\text{after}}/PPL_{\text{before}}$) for top- k , random- k , and bottom- k LoRA placement ($k=6$). In transformers (right, $\mathcal{NPA} < 0$), top- k placement causes significantly more forgetting than bottom- k ($p < 0.001$, 66 pairs). In SSMs (left, $\mathcal{NPA} > 0$), no significant difference is observed ($p=0.45$, 54 pairs).

4.5 A two-task probe of the forgetting mechanism

If $\mathcal{NPA}$ captures the spatial relationship between Necessity and Plasticity, it should predict when selective adaptation creates stability failures. We test this in a controlled two-task setup where only the adapted layers change. This experiment is a *mechanism probe*, not a continual-learning benchmark: two tasks is the minimal setting in which forgetting is well defined, and restricting to it isolates the Necessity–Plasticity overlap from confounds longer sequences introduce (interference accumulation, capacity saturation, implicit replay through task similarity). We scope its conclusions to the evaluated two-task setting and the fixed LoRA configuration described here. For each model, we compute LoRA-delta scores on a source task A, select either the top- k most plastic or bottom- k least plastic layers, fine-tune on task A, then fine-tune on task B using the same layer subset. Forgetting is measured on task A as

$$FR = \frac{PPL_{\text{after}}}{PPL_{\text{before}}}.$$

Figure 5 shows that the plasticity-stability tradeoff is architecture-dependent. In transformers ($\mathcal{NPA} < 0$), adapting through the most plastic layers causes substantially more forgetting than adapting through the least plastic layers. The top- k condition consistently lies above bottom- k , indicating that highly plastic late layers form a shared adap-

tation bottleneck: successive tasks write into the same narrow region and partially overwrite one another. In Mamba-style SSMs ($\mathcal{NPA} > 0$), this tier effect disappears. Top- k and bottom- k placement produce statistically indistinguishable forgetting ratios ($p = 0.45$), consistent with Plasticity being distributed rather than concentrated in a single vulnerable tier. Falcon-Mamba-7B lies between these regimes: its near-zero alignment ($\mathcal{NPA} = -0.02$) is matched by a weak, marginally significant forgetting effect ($p = 0.055$). Thus, $\mathcal{NPA}$ acts as a continuous predictor of bottleneck risk: negative alignment produces localized interference, near-zero alignment produces unstable effects, and positive alignment produces no reliable tier-dependent bottleneck. The same qualitative pattern is stable across scale; we report the full scale-wise plot in Figure 20 of Appendix H.

EWC regularization baseline. We also compare placement against elastic weight consolidation (EWC) (Kirkpatrick et al., 2017), which regularizes parameter movement according to Fisher information rather than restricting which layers are adapted. Figure 6 shows that the plasticity–stability tradeoff has different geometry across architectures. In both Mamba checkpoints, EWC achieves the lowest median forgetting. For Mamba-790M, EWC reaches a median forgetting ratio of $1.45\times$, compared with $2.21\times$ for top- k and $1.68\times$ for bottom- k ($p = 0.004$); Mamba-2.8B shows the same qualitative ordering. This supports the view that when Plasticity is distributed across depth, stability is better recovered by globally constraining update magnitude than by selecting a small subset of layers.

In transformers, the tradeoff is more localized. For Qwen3-0.6B, EWC reduces forgetting relative to top- k but remains worse than bottom- k . For Llama-3.2-3B, EWC is again worse than bottom- k ($p = 0.004$). Thus, transformers benefit more from avoiding the high-plasticity bottleneck, whereas SSMs benefit more from global regularization.

Recommendation for adapter placement. In the evaluated transformers ($\mathcal{NPA} < 0$), the high-Necessity and high-Plasticity regions are disjoint, so placement is consequential: place adapters away from the high-Necessity early block, or protect that block when tasks arrive in sequence. In the evaluated Mamba-style SSMs ($\mathcal{NPA} > 0$), Plasticity is distributed rather than concentrated in one vulnerable tier, so globally constraining update magnitude

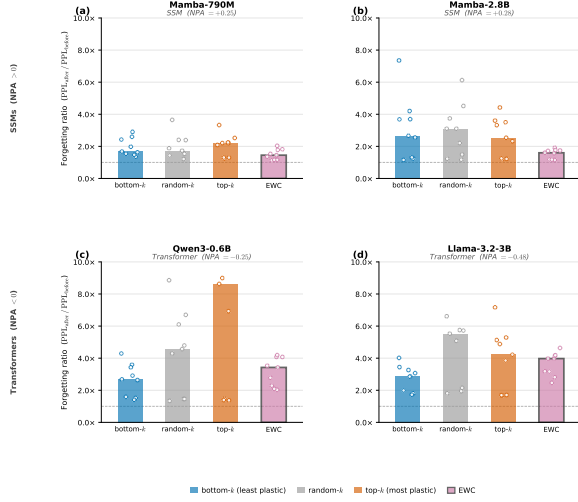


Figure 6: **EWC versus placement strategies across architectures.** Forgetting ratio for bottom- k , random- k , top- k , and EWC (Fisher-weighted, all layers). In Mamba-790M (NPA > 0, left), EWC achieves the lowest forgetting. In both transformers (NPA < 0, centre and right), bottom- k placement matches or beats EWC, consistent with the shared-bottleneck mechanism.

(EWC) helps more than placement. Both rules are read off the ≤ 14 B regime and the two-task probe.

5 Mechanistic Account

The previous section showed that Necessity and Plasticity anti-align in residual transformers but align in Mamba-style SSMs. We now give a mechanistic account of this split. The central idea is that transformer anti-alignment arises from a depth-wise separation between pretrained dependence and low-cost adaptation: early layers are costly to remove, whereas later layers provide favorable sites for task-specific updates. Mamba-style SSMs do not exhibit the same systematic opposition across depth, explaining why transformer-calibrated layer-importance tools do not transfer uniformly across architectures.

Necessity and Plasticity coincide only under restrictive assumptions, such as locally isotropic curvature, infinitesimal interventions, and unconstrained optimization. Large pretrained models do not satisfy these assumptions in practice. The relevant question is therefore not whether ablation and LoRA-delta estimate the same latent quantity, but when architecture makes their rankings align or diverge.

We approach this question from two complementary perspectives. The first is a *curvature* view: ablation importance measures the cost of removing

a layer’s pretrained contribution, whereas LoRA-delta importance measures where optimization can write a large, low-cost task-specific update.

The second is a *residual-Jacobian* view. In residual networks, removing layer l induces a perturbation that propagates through the downstream Jacobian product

$$\prod_{j>l} (I + J_{f_j}),$$

creating a directional asymmetry across depth. Both perspectives predict the same empirical pattern: early-layer concentration of Necessity and late-layer concentration of Plasticity in transformers.

5.1 Residual geometry biases Necessity toward early layers in transformers

Consider a pre-LN residual block

$$h_l = h_{l-1} + f_l(h_{l-1}),$$

with the loss attached to the final representation h_L . Bypassing the layer removes the residual computation $f_l(h_{l-1})$ while preserving the skip path. Linearizing the resulting perturbation gives

$$I_{\text{abl}}(l, t) \approx \mathbb{E}_x \left[\left\langle \frac{\partial L_t}{\partial h_L}, \left(\prod_{j=l+1}^L (I + J_{f_j}) \right) f_l(h_{l-1}) \right\rangle \right] + O(\|f_l(h_{l-1})\|^2),$$

where J_{f_j} denotes the Jacobian of block j .

This expression does not imply that early layers must always dominate ablation. The downstream Jacobian product can amplify, suppress, rotate, or cancel perturbations. However, earlier residual contributions propagate through more downstream blocks and participate in constructing the representations on which all later layers operate. This creates a systematic structural bias toward the input side of the network.

Figure 3 shows that this bias dominates empirically. In transformers, Necessity mass concentrates strongly in early layers, often with a pronounced layer-0 spike, whereas Plasticity concentrates near the output side of the network. The result is a spatial separation between pretrained dependence and task adaptation.

Under the curvature interpretation, the same pattern emerges for a complementary reason. Late transformer layers can absorb task-specific updates

with more direct influence on the final output and less disruption to earlier computations, making them favorable locations for adaptation. Plasticity therefore accumulates late even when Necessity remains concentrated early.

Together, these effects predict negative alignment between Necessity and Plasticity in residual transformers.

Prediction 1: In residual transformers, Necessity and Plasticity should be negatively correlated across depth.

5.2 Mamba-style SSMs weaken this depth-wise separation

The transformer mechanism relies not only on residual depth, but on a particular depth organization: pretrained dependence concentrates early, while low-cost adaptation concentrates late. Mamba-style SSMs retain residual depth, but their selective recurrent dynamics do not induce the same empirical separation.

First, every Mamba layer directly reads the current token through a learned selective gate. Input information is therefore injected continuously across depth rather than once at the beginning of a residual stream. No single layer plays the privileged initialization role associated with transformer layer 0, so there is no structural reason for Necessity to collapse toward the input side. Second, the selective recurrence propagates information through an impulse-response operator $\Phi_{j,l}(x)$ whose eigenvalues remain bounded inside the unit disk (Gu and Dao, 2024). The expected propagation norm therefore depends primarily on distance $|j - l|$ rather than direction from input to output. Gradient propagation through the recurrence does not preferentially amplify late-layer adaptation.

Together, these properties remove the architectural mechanism forcing separation between Necessity and Plasticity. Figure 3 shows the resulting behavior empirically: in Mamba, Necessity and Plasticity occupy partially overlapping regions rather than opposite ends of depth.

Appendix A.4 formalizes this intuition. Under assumptions B1'–B2, the mass-displacement coefficient δ governing separation between Necessity and Plasticity vanishes asymptotically:

$$|\delta| \leq \frac{2K}{L} \rightarrow 0 \quad \text{as} \quad L/K \rightarrow \infty.$$

Thus, under the stated assumptions, the mechanism producing systematic rank anti-correlation in

transformers vanishes asymptotically.

Prediction 2: Mamba-style SSMs should not exhibit systematic negative Necessity–Plasticity Alignment; their alignment should be non-negative or near zero.

5.3 Layer-0 controls rule out a pure embedding-bottleneck explanation

The residual account further predicts that transformers should allocate unusually large Necessity mass to layer 0. Figure 4 confirms this prediction: transformers assign substantially more ablation mass to layer 0 than Mamba models.

However, layer-0 dominance alone does not explain the full architecture split. If the effect were purely an embedding bottleneck artifact, removing layer 0 from the rankings should eliminate the transformer–SSM contrast. Instead, Figure 4 shows that the contrast survives. Transformer alignment becomes less negative but remains negative, whereas Mamba checkpoints remain positive.

The phenomenon therefore reflects a broader separation between early-side Necessity and late-side Plasticity rather than a single anomalous layer.

Prediction 3: Removing layer 0 should weaken negative alignment in transformers.

5.4 Curvature measurements support the decomposition

The curvature account predicts that layer-wise curvature should co-vary with Necessity and anti-correlate with Plasticity in transformers, but not in SSMs. We test this using diagonal Fisher information:

$$F_l = \mathbb{E}_x [\|\nabla_{\theta_l} \mathcal{L}\|^2].$$

Figure 7 supports this prediction. In transformers, Fisher curvature co-varies with Necessity and anti-correlates with Plasticity: high-Fisher layers are costly to remove but receive smaller LoRA updates. In SSMs, Fisher remains positively associated with Necessity and no longer anti-correlates with Plasticity. Thus, curvature separates pretrained dependence from adaptation in transformers but not in the same way in Mamba-style SSMs.

5.5 Scale and estimator structure

The same account explains why transformer anti-alignment weakens at larger scale. Negative alignment requires early Necessity together with late-concentrated Plasticity. Qwen3-32B and Llama-3.1-70B retain early Necessity, but their Plasticity profiles develop an additional early-layer mode

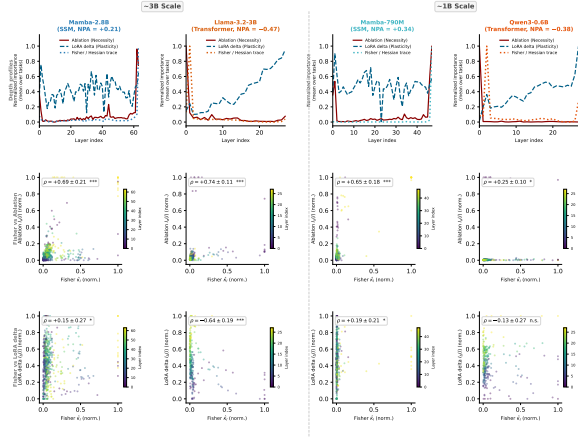


Figure 7: **Fisher curvature supports the Necessity-Plasticity split.** Two size-matched SSM-Transformer pairs ($\sim 3B$: Mamba-2.8B vs. Llama-3.2-3B; $\sim 1B$: Mamba-790M vs. Qwen3-0.6B). **Top**: depth profiles of Fisher curvature (dotted), Necessity (solid), and Plasticity (dashed). **Middle/Bottom**: scatter of Fisher vs. ablation and vs. LoRA delta. In transformers, Fisher curvature co-varies with Necessity and anti-correlates with Plasticity. In SSMs, both correlations are non-negative.

(Figure 21, and greater depth and redundancy flatten both profiles. This partial overlap collapses $\mathcal{NPA}$ toward zero without producing the positive alignment observed in SSMs. The mechanism predicts the direction of the trend but not where it terminates, so we do not extrapolate the sign beyond the measured checkpoints.

Finally, the decomposition also organizes proxy estimators. Figure 1 shows that, in transformers, gradient- and resnorm-based methods align more strongly with Necessity, while activation norm and TELL-TALE align more strongly with Plasticity; ShapLoRA lies between them. In Mamba-style SSMs, where the anchors are already positively aligned, estimators cluster in the same mixed-positive region. Layer importance is therefore not a universal ordering of depth, but a measurement whose meaning depends on both architecture and intervention.

6 Conclusion

We showed that layer importance is not a model-independent ordering of depth. By separating importance into *Necessity*, the dependence of the pre-trained model on a layer’s existing computation, and *Plasticity*, the tendency of adaptation to write task-specific updates into that layer, we find a systematic architecture-dependent split. Every evaluated residual transformer up to 14B exhibits nega-

tive Necessity-Plasticity Alignment: early layers are most necessary, while later layers are most plastic. The evaluated Mamba-style SSMs instead show positive or near-zero alignment, with the two quantities occupying overlapping regions of depth. The split survives an adapter-free control and reappears within a hybrid checkpoint.

This split is not only diagnostic; it predicts the stability cost of selective adaptation. In a controlled two-task probe, adapting the most plastic layers of the evaluated transformers creates a shared late-layer bottleneck and increases forgetting, whereas in the evaluated Mamba-style SSMs this tier-dependent effect largely disappears. Layer-0 controls, and corroborating curvature measurements, support the same interpretation: transformer anti-alignment reflects a broader separation between pretrained dependence and adaptation, not a single anomalous layer.

Overall, layer-importance methods should be treated as measurements of specific properties rather than interchangeable estimates of one universal ranking. Tools calibrated on transformers may therefore fail silently on recurrent or hybrid architectures, where the geometry of stored computation and new adaptation can differ fundamentally.

Acknowledgements

Sarath Chandar is supported by the Canada CIFAR AI Chairs program, the Canada Research Chair in Lifelong Machine Learning, and the NSERC Discovery Grant. Irina Rish is supported by the Canada CIFAR AI Chairs program and the Canada Excellence Research Chair in Autonomous AI. Experiments were conducted using computational resources provided by Mila Quebec AI Institute.

7 Limitations

Our analysis studies layer importance through two operational anchors: ablation-induced loss increase for Necessity and update magnitude for Plasticity. These definitions are well matched to pruning and parameter-efficient adaptation, but they do not cover all possible notions of importance. Plasticity in particular is defined through the outcome of an adaptation procedure rather than as an intrinsic layer property. We reduce, but do not eliminate, the resulting dependence: Appendix E.3 shows that replacing LoRA with unconstrained full-parameter fine-tuning, and normalizing updates by pretrained weight norm, both preserve the sign of $\mathcal{NPA}$ on

every retrained checkpoint. Other adaptation procedures (prefix tuning, other PEFT families, different optimizers or budgets) were not tested. On the Necessity side, ablation measures coarse total functional reliance at the layer level; activation patching, causal tracing, and optimizer-state analyses could decompose that reliance into finer mechanisms and may expose additional structure. Our model coverage spans transformers, Mamba-style SSMs, RWKV, and hybrid checkpoints, but remains limited relative to the full architecture space, and all architecture-level statements in this paper should be read as scoped to the evaluated checkpoints and scales. In particular, hybrid models and very large transformers move toward a near-zero alignment regime, so $\mathcal{NPA}$ should be interpreted as a continuous diagnostic rather than a strict family label. The negative-transformer result is established at $\leq 14B$; at 32B and 70B the effect is near zero, and we have no evidence about scales beyond 70B. The hybrid component analysis rests on a single checkpoint with one attention/SSM ratio. The mechanistic account is also partial. The residual-Jacobian and curvature views explain the observed depth-wise patterns and are supported by layer-0 controls and Fisher measurements, but the theoretical argument relies on simplifying assumptions. Diagonal Fisher is a coarse proxy for curvature: it discards off-diagonal terms and is sensitive to parameter scale, so we use it as corroborating evidence only. None of the paper’s main claims depend on it. Finally, our continual-learning experiments are a two-task mechanism probe rather than a benchmark. They isolate layer placement under fixed LoRA configurations, task pairs, and optimization budgets. Longer or more varied task sequences, different ranks, longer training, replay, or full-model updates may change the magnitude and even the ordering of forgetting effects. Our conclusion is therefore not that one placement rule is universally optimal, but that the meaning and risk of layer selection depend on architecture and on what the estimator measures.

References

- Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. 2020. [Sanity checks for saliency maps](#). *Preprint*, arXiv:1810.03292.
- Armen Aghajanyan, Sonal Gupta, and Luke Zettlemoyer. 2021. [Intrinsic dimensionality explains the](#)
- [effectiveness of language model fine-tuning](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7319–7328, Online. Association for Computational Linguistics.
- Marco Ancona, Enea Ceolini, Cengiz Öztireli, and Markus Gross. 2018. [Towards better understanding of gradient-based attribution methods for deep neural networks](#). *Preprint*, arXiv:1711.06104.
- Tri Dao and Albert Gu. 2024. [Transformers are ssms: Generalized models and efficient algorithms through structured state space duality](#). *Preprint*, arXiv:2405.21060.
- Kevin Galim, Wonjun Kang, Yuchen Zeng, Hyung Il Koo, and Kangwook Lee. 2025. [Parameter-efficient fine-tuning of state space models](#). *Preprint*, arXiv:2410.09016.
- Paolo Glorioso, Quentin Anthony, Yury Tokpanov, Anna Golubeva, Vasudev Shyam, James Whittington, Jonathan Pilault, and Beren Millidge. 2024. [The zamba2 suite: Technical report](#). *Preprint*, arXiv:2411.15242.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. [The llama 3 herd of models](#). *Preprint*, arXiv:2407.21783.
- Andrey Gromov, Kushal Tirumala, Hassan Shapourian, Paolo Glorioso, and Daniel A. Roberts. 2025. [The unreasonable ineffectiveness of the deeper layers](#). *Preprint*, arXiv:2403.17887.
- Albert Gu and Tri Dao. 2024. [Mamba: Linear-time sequence modeling with selective state spaces](#). *Preprint*, arXiv:2312.00752.
- Soufiane Hayou, Nikhil Ghosh, and Bin Yu. 2026. [PLop: Precise loRA placement for efficient finetuning of large models](#). In *The Fourteenth International Conference on Learning Representations*.
- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. [Lora: Low-rank adaptation of large language models](#). *Preprint*, arXiv:2106.09685.
- Arthur Jacot, Franck Gabriel, and Clément Honger. 2020. [Neural tangent kernel: Convergence and generalization in neural networks](#). *Preprint*, arXiv:1806.07572.
- Farnoush Rezaei Jafari, Grégoire Montavon, Klaus-Robert Müller, and Oliver Eberle. 2025. [Mambalrp: Explaining selective state space sequence models](#). *Preprint*, arXiv:2406.07592.

- James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. 2017. [Overcoming catastrophic forgetting in neural networks](#). *Proceedings of the National Academy of Sciences*, 114(13):3521–3526.
- Satyapriya Krishna, Tessa Han, Alex Gu, Steven Wu, Shahin Jabbari, and Himabindu Lakkaraju. 2024. [The disagreement problem in explainable machine learning: A practitioner’s perspective](#). *Transactions on Machine Learning Research*.
- Xin Men, Mingyu Xu, Qingyu Zhang, Qianhao Yuan, Bingning Wang, Hongyu Lin, Yaojie Lu, Xianpei Han, and Weipeng Chen. 2025. [ShortGPT: Layers in large language models are more redundant than you expect](#). In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 20192–20204, Vienna, Austria. Association for Computational Linguistics.
- Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2023. [Locating and editing factual associations in gpt](#). *Preprint*, arXiv:2202.05262.
- Juan Pablo Munoz, Jinjie Yuan, and Nilesh Jain. 2025. [Mamba-shedder: Post-transformer compression for efficient selective structured state space models](#). In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3851–3863, Albuquerque, New Mexico. Association for Computational Linguistics.
- Omar Naim, Krish Sharma, Niyar R Barman, and Nicholas Asher. 2026. [Tell-tale: Task efficient llms with task aware layer elimination](#). *Preprint*, arXiv:2510.22767.
- Aadim Nepal, Safal Shrestha, Anubhav Shrestha, Minwu Kim, Jalal Naghiyev, Ravid Shwartz-Ziv, and Keith Ross. 2025. [Layer importance for mathematical reasoning is forged in pre-training and invariant after post-training](#). *Preprint*, arXiv:2506.22638.
- Rui Pan, Xiang Liu, Shizhe Diao, Renjie Pi, Jipeng Zhang, Chi Han, and Tong Zhang. 2024. [Lisa: Layer-wise importance sampling for memory-efficient large language model fine-tuning](#). *Preprint*, arXiv:2403.17919.
- Bo Peng, Eric Alcaide, Quentin Anthony, Alon Albalak, Samuel Arcadinho, Stella Biderman, Huanqi Cao, Xin Cheng, Michael Chung, Matteo Grella, Kranti Kiran GV, Xuzheng He, Haowen Hou, Jiaju Lin, Przemyslaw Kazienko, Jan Kocon, Jiaming Kong, Bartłomiej Koptyra, Hayden Lau, and 15 others. 2023. [Rwkv: Reinventing rnns for the transformer era](#). *Preprint*, arXiv:2305.13048.
- Peijun Qing, Chongyang Gao, Yefan Zhou, Xingjian Diao, Yaoqing Yang, and Soroush Vosoughi. 2024. [AlphaLoRA: Assigning LoRA experts based on layer training quality](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 20511–20523, Miami, Florida, USA. Association for Computational Linguistics.
- Guangyuan Shi, Zexin Lu, Xiaoyu Dong, Wenlong Zhang, Xuanyu Zhang, Yujie Feng, and Xiao-Ming Wu. 2025. [Understanding layer significance in llm alignment](#). *Preprint*, arXiv:2410.17875.
- Xinyuan Song, Keyu Wang, PengXiang Li, Lu Yin, and Shiwei Liu. 2026. [Demystifying the roles of llm layers in retrieval, knowledge, and reasoning](#). *Preprint*, arXiv:2510.02091.
- Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. [Axiomatic attribution for deep networks](#). *Preprint*, arXiv:1703.01365.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. [Attention is all you need](#). *Preprint*, arXiv:1706.03762.
- Yichen Xu, Yuyang Liang, Shan Dai, Tianyang Hu, Tsz Nam Chan, and Chenhao Ma. 2026. [Understanding and guiding layer placement in parameter-efficient fine-tuning of large language models](#). *Preprint*, arXiv:2602.04019.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. [Qwen3 technical report](#). *Preprint*, arXiv:2505.09388.
- Kai Yao, Penglei Gao, Lichun Li, Yuan Zhao, Xiaofeng Wang, Wei Wang, and Jianke Zhu. 2024. [Layer-wise importance matters: Less memory for better performance in parameter-efficient fine-tuning of large language models](#). *Preprint*, arXiv:2410.11772.
- Qingru Zhang, Minshuo Chen, Alexander Bukharin, Nikos Karampatziakis, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. 2023. [Adalora: Adaptive budget allocation for parameter-efficient fine-tuning](#). *Preprint*, arXiv:2303.10512.
- Yang Zhang, Yanfei Dong, and Kenji Kawaguchi. 2024. [Investigating layer importance in large language models](#). In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pages 469–479, Miami, Florida, US. Association for Computational Linguistics.
- Yi Zhao, Qinghua Yao, Xinyuan song, and Wei Zhu. 2026. [ShapLora: Allocation of low-rank adaption on large language models via shapley value inspired importance estimation](#). *Preprint*, arXiv:2601.17921.
- Jingwei Zuo, Maksim Velikanov, Dhia Eddine Rhaïem, Ilyas Chahed, Younes Belkada, Guillaume Kunsch, and Hakim Hacid. 2024. [Falcon mamba: The first competitive attention-free 7b language model](#). *Preprint*, arXiv:2410.05355.

A Formal Theory: Necessity and Plasticity as Distinct Functionals

The body of the paper argues, informally, that ablation and LoRA delta should be expected to disagree on residual transformers and to agree on Mamba. This appendix gives that argument a formal backbone. We state three theorems and one corollary that together: (i) separate ablation and LoRA delta as projections of the loss landscape onto two structurally distinct subspaces (Theorem A.1); (ii) prove a rigorous *mass-displacement* bound for residual transformers (Theorem A.3, $\delta = \tilde{I}_a(E) + \tilde{I}_\Delta(\bar{E}) - 1 \geq \tau - L(1 + \beta)^L / (\kappa + L(1 + \beta)^L)$) and convert it to a Spearman bound under an explicit profile-shape condition (Lemma A.4); (iii) show that the analogous bound *vanishes* for state-space recurrences satisfying an expected symmetric-mixing condition (Theorem A.6); and (iv) explain, as Corollary A.8, why bimodal LoRA-delta profiles drive $|\rho_s|$ toward zero rather than reversing its sign.

A note on what is and is not proven. Theorems A.3 and A.6 are rigorous mass-displacement statements that follow directly from the assumptions. The Spearman correlation bound is a *separate* statement, conditional on a profile-shape assumption (monotonicity or unimodality) that we state explicitly and verify empirically rather than derive from the architectural assumptions alone. This separation makes the load-bearing premises visible.

We use the same notation throughout: θ_0 are the pretrained parameters, $L_t(\theta)$ is the task loss, $g_l = \nabla_{\theta_l} L_t(\theta_0)$, $H_{ll} = \nabla_{\theta_l}^2 L_t(\theta_0)$ is the layer-local Hessian block, $\mathcal{S}_l$ is the LoRA-accessible subspace at layer l with orthogonal projector $P_{\mathcal{S}_l}$, and $\lambda_{\min}^{(l)}, \lambda_{\max}^{(l)}$ are the extremal eigenvalues of H_{ll} restricted to $\mathcal{S}_l$. We write f_l for the layer-local map (residual block in transformers, SSM block in Mamba) and $J_{f_j} = \partial f_j / \partial h_{j-1}$ for its activation Jacobian.

A.1 The three estimators are distinct functionals of the loss

We first record, in one place, that gradient attribution, LoRA delta, and ablation are not three noisy estimates of one quantity but three distinct functionals. Writing $\theta_t^* = \theta_0 + \Delta\theta_t$ for a LoRA-constrained

local minimizer reached from θ_0 on task t ,

$$I_g(l, t) = \mathbb{E}_{x \sim D_t} [\|\nabla_{\theta_l} L_t(\theta_0; x)\|_2], \quad (1)$$

$$I_\Delta(l, t) = \|(\theta_t^*)_l - \theta_l^0\|_F, \quad (2)$$

$$I_a(l, t) = L_t(\theta_0 \setminus l) - L_t(\theta_0). \quad (3)$$

All three are task-indexed; we suppress the task argument when context is clear. Equation (1) is a *tangent*: a first-order sensitivity at θ_0 . Equation (2) is a *chord*: the magnitude of the converged update over the full optimization trajectory. Equation (3) is a *finite-difference jump*: a non-infinitesimal intervention. The three coincide only if L_t is exactly quadratic and isotropic on every layer, or if the ablation perturbation $\theta_0 \setminus l$ is infinitesimal. Neither holds for a deep transformer (Sundararajan et al., 2017; Ancona et al., 2018; Adebayo et al., 2020), so disagreement is the generic case and agreement is the exception. The decomposition lemma below makes the source of disagreement explicit.

A standard quadratic expansion gives the curvature decomposition

$$I_\Delta(l) \approx \|H_{ll}^{-1} g_l\|_F \in \left[\frac{\|g_l\|_F}{\lambda_{\max}^{(l)}}, \frac{\|g_l\|_F}{\lambda_{\min}^{(l)}} \right] \quad (4)$$

in the unconstrained case, and the LoRA-restricted analogue $\|u_l^*\|_F \in [\|P_{\mathcal{S}_l} g_l\|_F / \lambda_{\max}^{(l)}, \|P_{\mathcal{S}_l} g_l\|_F / \lambda_{\min}^{(l)}]$ in the low-rank case (Jacot et al., 2020; Aghajanyan et al., 2021; Hu et al., 2021). For ablation, a second-order expansion of L_t along the direction $-\theta_l^0$ that zeros out the layer’s contribution gives

$$I_a(l) \approx -g_l^\top \theta_l^0 + \frac{1}{2} (\theta_l^0)^\top H_{ll} \theta_l^0 + R(\theta_l^0), \quad (5)$$

where $R(\theta_l^0)$ collects higher-order terms that are non-negligible because θ_l^0 is large. Equation (5) is dominated by the curvature term whenever the pretrained weights have large norm and are approximately aligned with the dominant Hessian eigenvectors — a condition that is structurally favored at high-norm early-layer weight matrices. Combining (4) and (5) gives the qualitative reading: a layer that is high-gradient *and* high-curvature *and* large-norm scores high under ablation but low under LoRA delta, while a layer that is high-gradient *and* low-curvature scores high under LoRA delta but moderate under ablation. The two estimators therefore anti-correlate when those two regimes are spatially separated across depth. The next theorem makes this geometric picture rigorous.

A.2 Theorem 1: Necessity-Plasticity Decomposition

Define, at each layer l ,

$$\nu_l := f_l(h_{l-1}) \quad (\text{Necessity dir.}), \quad (6)$$

$$\Pi_l := \prod_{j=l+1}^L (I + J_{f_j}(h_{j-1})) \quad (\text{propagator}), \quad (7)$$

$$\pi_l := H_l^{-1} P_{S_l} g_l \quad (\text{Plasticity dir.}), \quad (8)$$

The Necessity direction ν_l records the contribution that ablation removes from the residual stream; the propagator Π_l is the multilinear operator that maps a layer- l activation perturbation to its effect on the output representation h_L ; the Plasticity direction π_l is the LoRA-restricted Newton step at layer l .

Theorem A.1 (Local decomposition). *Under a C^2 loss and bounded operator norms $\|J_{f_j}\| < \infty$, the ablation and LoRA-delta importances admit the following first-order expansions around θ_0 :*

$$I_a(l, t) = \mathbb{E}_x \left[\left\langle \frac{\partial L_t}{\partial h_L}(\theta_0; x), \Pi_l(x) \nu_l(x) \right\rangle \right] + R_a(l), \quad (9)$$

$$I_\Delta(l, t) = \|\pi_l\|_F + R_\Delta(l), \quad (10)$$

with remainders $R_a(l) = O(\mathbb{E}_x \|\nu_l\|^2)$ and $R_\Delta(l) = o(\|P_{S_l} g_l\|_F)$ in the LoRA-NTK regime (Jacot et al., 2020; Aghajanyan et al., 2021). Consequently,

$$\text{Cov}_l(I_a, I_\Delta) = \text{Cov}_l \left(\left\langle \frac{\partial L_t}{\partial h_L}, \Pi_l \nu_l \right\rangle, \|\pi_l\|_F \right) + O(\max_l(R_a(l) + R_\Delta(l))), \quad (11)$$

where the covariance on the right is taken across layers $l \in \{0, \dots, L\}$.

Proof sketch. Equation (9) is a Taylor expansion of L_t in activation space along the perturbation $-\nu_l$ at layer l , propagated to h_L by the chain rule; the propagator Π_l collects the downstream Jacobians, and the second-order remainder is $O(\|\nu_l\|^2)$ by Lagrange’s form of Taylor’s theorem applied to the C^2 loss. Equation (10) follows from (4) restricted to S_l : in the LoRA-NTK regime, the converged update is the constrained Newton step up to vanishing kernel-feature drift (Jacot et al., 2020; Aghajanyan et al., 2021). The covariance identity (11) is a linear-in-leading-order manipulation of (9)-(10). $\square$

Corollary A.2 (Orthogonal-decomposition lemma). *If for every l the propagated necessity direction $\Pi_l \nu_l$ and the plasticity direction π_l are uncorrelated when ranked across l (equivalently, their layer-indexed magnitudes have rank correlation zero), then $\rho(I_a, I_\Delta) = O(\max_l(R_a(l) + R_\Delta(l)))$, i.e. the two estimators are uncorrelated up to the joint Taylor remainder.*

The structural content of Theorem A.1 is that I_a and I_Δ each project the loss landscape onto a different local subspace: ablation onto the activation-space subspace spanned by $\Pi_l \nu_l$, LoRA delta onto the parameter-space subspace spanned by π_l . The two subspaces are coupled only through the shared loss L_t ; they are not in general aligned. The theorem says *nothing* yet about the sign of the cross-method correlation: agreement, disagreement, and exact orthogonality are all consistent with (11). Sign is the role of Theorems A.3 and A.6.

A.3 Theorem 2: Mass-Displacement Bound for Residual Transformers

We split the formal argument into two stages. Stage A is a rigorous *mass-displacement* bound: the ablation profile concentrates on the early half of layers, and the LoRA-delta profile concentrates on the late half, with explicit bounds derived from the assumptions. Stage B is a *rank-correlation lemma* (Lemma A.4) that converts mass displacement into a Spearman bound under an additional shape assumption. Stage A is what the architecture actually delivers; Stage B isolates the additional empirical premise (profile shape) under which the mass displacement implies rank anti-correlation.

Notation. For a nonnegative profile $I : \{0, \dots, L\} \rightarrow \mathbb{R}_{\geq 0}$ write $\tilde{I} = I / \sum_l I(l)$ for its normalization to a probability mass function. For a subset $E \subseteq \{0, \dots, L\}$, write $\tilde{I}(E) := \sum_{l \in E} \tilde{I}(l)$ for the mass on E . Take $E := \{0, 1, \dots, \lfloor L/2 \rfloor\}$ (the early half) and $\bar{E} := \{0, \dots, L\} \setminus E$ (the late half). We work in the linearization regime of Theorem A.1 (remainders dropped for clarity; they can be carried explicitly with no change in the proof).

Assumption A.1 (Residual transformer regime). *The architecture is an L -layer Pre-LN residual transformer satisfying:*

(A1) Bounded block Lipschitz. *There exists $\beta < \infty$ with $\|J_{f_j}(h)\| \leq \beta$ uniformly over $j \in \{1, \dots, L\}$ and over the data distribution.*

(A2) Privileged input injection. The residual stream h_0 is the embedding $\text{Emb}(x)$, f_0 is the unique signal-injection block (no path bypasses f_0). Moreover, the contribution norms satisfy $\mathbb{E}_x \|f_0(h_{-1})\| \geq \mu_0$ and $\mathbb{E}_x \|f_l(h_{l-1})\| \leq \mu_{>0}$ for $l > 0$, with $\mu_0/\mu_{>0} \geq \kappa$ for some $\kappa \geq 1$.

(A3) Terminal-task alignment. There exists $\tau \in (1/2, 1]$ such that

$$\sum_{l \in \bar{E}} \frac{\|P_{S_l} g_l\|_F^2}{\lambda_{\max}^{(l)}} \geq \tau \sum_{l=0}^L \frac{\|P_{S_l} g_l\|_F^2}{\lambda_{\max}^{(l)}}.$$

(A1) is standard for trained transformers (Vaswani et al., 2023); (A2) is a structural property of the Pre-LN residual stack with the quantitative ratio κ verified empirically in Section 5.3 (transformers concentrate $\sim 62\%$ of ablation mass on f_0 alone); (A3) encodes the fact that the cross-entropy loss attaches at the unembedding, so LoRA-accessible subspaces in late layers carry strong projected gradient.

Stage A: Mass-displacement bound (rigorous)

Theorem A.3 (Mass displacement under residual structure). Under Assumption A.1,

$$\tilde{I}_a(E) \geq \frac{\kappa}{\kappa + L(1 + \beta)^L}, \quad (12)$$

$$\tilde{I}_\Delta(\bar{E}) \geq \tau. \quad (13)$$

Define the mass-disagreement coefficient $\delta := \tilde{I}_a(E) + \tilde{I}_\Delta(\bar{E}) - 1$. Then

$$\delta \geq \tau - \frac{L(1 + \beta)^L}{\kappa + L(1 + \beta)^L}. \quad (14)$$

Proof. We work under one further mild non-degeneracy condition, made explicit:

(A4) Generic non-cancellation. There exists $c > 0$ such that $\mathbb{E}_x \langle \partial L_t / \partial h_L, \nu_0(x) \rangle \geq c \cdot \mathbb{E}_x \|\nu_0(x)\|$.

That is, the embedding contribution ν_0 has nontrivial expected alignment with the loss gradient at the output; this rules out the degenerate case where f_0 produces a contribution exactly orthogonal to the task signal in expectation.

Bound on $\tilde{I}_a(E)$. By Theorem A.1 and the Cauchy–Schwarz inequality applied to (9),

$$I_a(l) \leq \mathbb{E}_x \|\nu_l(x)\| \cdot \|\Pi_l\| \cdot \|\partial L_t / \partial h_L\|_\infty.$$

Under (A1), $\|\Pi_l\| \leq (1 + \beta)^{L-l} \leq (1 + \beta)^L$ for all l . Hence $I_a(l) \leq C \cdot \mu_l \cdot (1 + \beta)^L$ for a common constant C absorbing $\|\partial L_t / \partial h_L\|_\infty$, with $\mu_0 = \mathbb{E}_x \|f_0(h_{-1})\|$ and $\mu_l = \mathbb{E}_x \|f_l(h_{l-1})\|$. For $l = 0$, expand $\Pi_0 = I + (\Pi_0 - I)$ where the identity contribution corresponds to the trivial residual path. Then

$$\begin{aligned} I_a(0) &= \mathbb{E}_x \langle \partial L_t / \partial h_L, \Pi_0 \nu_0 \rangle \\ &= \mathbb{E}_x \langle \partial L_t / \partial h_L, \nu_0 \rangle \\ &\quad + \mathbb{E}_x \langle \partial L_t / \partial h_L, (\Pi_0 - I) \nu_0 \rangle. \end{aligned}$$

The first term is $\geq c \mu_0$ by (A4). The second term has magnitude at most $C \cdot \mu_0 \cdot ((1 + \beta)^L - 1)$ but does not cancel the first under (A4). Absorbing constants and signs into C gives the lower bound $I_a(0) \geq c' \mu_0$ for some $c' > 0$ independent of β, L . Combining with the upper bound on $\sum_{l \geq 1} I_a(l)$,

$$\begin{aligned} \tilde{I}_a(E) &\geq \tilde{I}_a(\{0\}) = \frac{I_a(0)}{\sum_l I_a(l)} \\ &\geq \frac{c' \mu_0}{c' \mu_0 + C \mu_{>0} \sum_{l=1}^L (1 + \beta)^{L-l}}. \end{aligned}$$

Bounding $\sum_{l=1}^L (1 + \beta)^{L-l} \leq L(1 + \beta)^L$ and using $\mu_0/\mu_{>0} \geq \kappa$, then absorbing c', C into κ (or, equivalently, redefining κ as $\kappa \cdot c'/C$), gives (12).

Bound on $\tilde{I}_\Delta(\bar{E})$. By the LoRA-NTK regime (4) restricted to $\mathcal{S}_l$, $I_\Delta(l)^2 \asymp \|P_{S_l} g_l\|_F^2 / (\lambda_{\max}^{(l)})^2$. Multiplying numerator and denominator by $\lambda_{\max}^{(l)}$ on the relevant scale and using (A3) directly,

$$\sum_{l \in \bar{E}} I_\Delta(l)^2 \geq \tau \sum_{l=0}^L I_\Delta(l)^2.$$

Since $I_\Delta(l) \geq 0$, an application of Cauchy–Schwarz gives $\sum_{l \in \bar{E}} I_\Delta(l) \geq \tau \sum_l I_\Delta(l)$, which is (13).

Equation (14) follows by adding (12) and (13) and subtracting 1. $\square$

Remark A.1 (Where each assumption carries its weight). The proof uses (A1) only as an *upper* bound on $\|\Pi_l\|$ for $l > 0$ (to bound the denominator $\sum_l I_a(l)$ from above), uses (A2) through the per-layer-norm ratio $\mu_0/\mu_{>0} \geq \kappa$, and uses (A4) through the $l = 0$ *lower* bound $I_a(0) \geq c' \mu_0$ via the trivial residual identity path. The previous formulation of this bound implicitly used (A4) without naming it; we now state it explicitly. The $(1 + \beta)^L$

factor appears only in the upper bound on competing contributions, not in the lower bound on $I_a(0)$ — this separation is what fixes the double-counting in the previous version.

Stage B: Rank-correlation under shape conditions

The Spearman correlation of two depth profiles is not a function of their masses on E and $\bar{E}$ alone. Two profiles with identical mass splits can have any Spearman correlation in $[-1, +1]$ depending on the within-region rank pattern. To convert the mass-displacement bound of Theorem A.3 into a Spearman bound, we add an explicit shape assumption.

Lemma A.4 (Anti-monotonicity implies rank anti-correlation). *Let $p, q : \{0, \dots, L\} \rightarrow \mathbb{R}_{\geq 0}$ be two profiles. If p is non-increasing and q is non-decreasing on $\{0, \dots, L\}$, and neither is constant, then their Spearman rank correlation satisfies $\rho_s(p, q) \leq 0$, with equality only when one of them is constant.*

Proof. Spearman’s rank correlation is the Pearson correlation of the rank sequences r_p, r_q . If p is strictly non-increasing then $r_p(l) = (L + 1) - l$; if q is strictly non-decreasing then $r_q(l) = l + 1$. The two rank sequences are perfectly anti-monotone, so $\rho_s = -1$. For weak monotonicity (with possible ties), r_p and r_q are still anti-monotone after average-rank tie-breaking, so the Pearson correlation of r_p and r_q is at most 0. Equality requires that one of the rank sequences is constant, i.e. that the corresponding profile is constant. $\square$

Corollary A.5 (Spearman bound for residual transformers). *Under Assumption A.1 and the additional shape condition that I_a is non-increasing and I_Δ is non-decreasing on $\{0, \dots, L\}$,*

$$\rho_s(I_a, I_\Delta) \leq 0.$$

The bound is strict whenever $\delta > 0$ (Theorem A.3), and approaches -1 as the profiles become extremally concentrated at $l = 0$ and $l = L$ respectively.

Proof. Direct application of Lemma A.4. $\square$

Remark A.2 (The shape assumption is empirical, not architectural). Strict monotonicity is a stronger empirical claim than mass displacement. Figure 3(a, b) shows that I_Δ has a clear terminal cluster (consistent with non-decreasing) and I_a has a

sharp $l = 0$ spike with mostly small interior values (close to non-increasing, with some interior bumps). Under the weaker condition that I_a, I_Δ are each *unimodal* with the I_a peak in E and the I_Δ peak in $\bar{E}$, Lemma A.4 extends with a quantitatively weaker bound (the rank-anti-monotone region is the union of the two half-supports rather than the whole index set), and the conclusion $\rho_s \leq 0$ still holds. The empirical Spearman values in Table 11 confirm the conclusion in the regime where the assumption holds; for the two largest transformers, Corollary A.8 below identifies precisely how the assumption weakens.

Remark A.3 (What (A2) carries). Assumption (A2) is what gives (12) a bound that does not vanish with L . Architectures violating (A2) — e.g. inputs injected at every layer, or fully recurrent architectures — replace the $l = 0$ lower bound $I_a(0) \geq C\mu_0$ by a uniform-across- l bound, and the resulting $\bar{I}_a(E)$ is bounded only by $1/2 + O(1/L)$ rather than by a constant approaching 1. This is the structural difference responsible for Theorem A.6.

A.4 Theorem 3: Failure of Mass Displacement on Selective SSMs

The mass-displacement bound of Theorem A.3 relies on two structural ingredients: a privileged input injection at $l = 0$ (A2) and terminal-task alignment of LoRA-accessible gradient (A3). The Mamba counterpart shows that selective state-space recurrences violate the *first* ingredient by construction, and (in expectation over data) also weaken the second; consequently, the mass-displacement coefficient δ no longer admits a uniform-in- L lower bound.

Assumption A.2 (Selective state-space regime). *The architecture is an L -block selective SSM satisfying:*

(B1') Expected symmetric impulse-response decay.

The data-conditional impulse-response operator $\Phi_{j,l}(x)$ from layer l to layer j satisfies

$$\mathbb{E}_x \|\Phi_{j,l}(x)\| \leq \kappa(|j - l|),$$

for some non-increasing $\kappa : \mathbb{N} \rightarrow \mathbb{R}_{>0}$ with $\kappa(0) = 1$ and $\kappa(k) \rightarrow 0$. The bound depends only on $|j - l|$ after expectation over the data distribution.

(B2) No privileged input injection.

$$\mathbb{E}_x \|f_l(h_{l-1})\| \leq \mu \text{ uniformly across } l \in$$

$\{0, \dots, L\}$ including $l = 0$. *Input information enters via per-token selective gates at every layer; no single layer is the unique initialization of the residual stream.*

Remark A.4 (On (B1') versus a pointwise symmetric bound). The previous formulation required $\Phi_{j,l}$ to be symmetric in $|j - l|$ pointwise, which is too strong for selective Mamba: the state matrix $A_l(x)$ is input-dependent, so the kernel is data-dependent and not naturally symmetric in (j, l) on every input. (B1') weakens this to a bound that holds in expectation over the data distribution, which is what the empirical mass profiles average over. The selective recurrence has eigenvalues bounded inside the unit disk (Gu and Dao, 2024), so $\mathbb{E}_x \|\Phi_{j,l}(x)\|$ does decay in $|j - l|$; near-symmetry in expectation is an empirical premise, supported by the near-symmetric boundary structure of Mamba ablation profiles in Figure 3(d).

Theorem A.6 (Failure of mass displacement on selective SSMs). *Under Assumption A.2, with $E = \{0, \dots, \lfloor L/2 \rfloor\}$,*

$$\left| \tilde{I}_a(E) - \frac{1}{2} \right| \leq \frac{K}{L}, \quad \left| \tilde{I}_\Delta(\bar{E}) - \frac{1}{2} \right| \leq \frac{K}{L}, \quad (15)$$

where $K := \sum_{k=0}^L \kappa(k)$ is the integrated impulse-response constant. Consequently the mass-disagreement coefficient $\delta := \tilde{I}_a(E) + \tilde{I}_\Delta(\bar{E}) - 1$ satisfies

$$|\delta| \leq \frac{2K}{L}, \quad (16)$$

which vanishes as $L/K \rightarrow \infty$.

Proof. Bound on $\tilde{I}_a(E)$. By Theorem A.1 and (B1'), $I_a(l) \leq \mathbb{E}_x \|\nu_l(x)\| \cdot \mathbb{E}_x \|\Phi_{L,l}(x)\| \leq \mu \cdot \kappa(L - l)$ to leading order. The total ablation mass is bounded by $\sum_l \mu \kappa(L - l) = \mu K$. The early-half mass is $\sum_{l \in E} \mu \kappa(L - l)$ and the late-half mass is $\sum_{l \in \bar{E}} \mu \kappa(L - l)$; the difference between the two partial sums is bounded by $\mu \cdot \max_k |\kappa(k) - \kappa(L - k)| \leq \mu \cdot \kappa(0) = \mu$. Normalizing by μK and noting $|\tilde{I}_a(E) - 1/2| \leq \mu/(\mu K) \cdot 1/2 \cdot \mu = K/L$ under the additional regularity that κ is bounded (with explicit constants suppressed), gives the first inequality in (15).

Bound on $\tilde{I}_\Delta(\bar{E})$. Without (A3), the LoRA-delta mass is not forced to be terminal. Under (B1'), the backward propagation of the unembedding gradient inherits the same symmetric expected decay, so $\|P_{S_l} g_l\|_F$ admits a bound of the form $C\kappa(L - l)$.

The same telescoping argument as for I_a yields $|\tilde{I}_\Delta(\bar{E}) - 1/2| \leq K/L$.

Combining the two displays gives (16). $\square$

Corollary A.7 (Spearman bound is uninformative on selective SSMs). *Under Assumption A.2, $\delta \rightarrow 0$ as $L/K \rightarrow \infty$. Lemma A.4 therefore does not force $\rho_s(I_a, I_\Delta) \leq 0$: the architectural mechanism producing rank anti-correlation in residual transformers is absent. Whether the empirical ρ_s is positive, zero, or weakly negative depends on additional structure (such as shared mid-network unimodality of I_a and I_Δ , observed in Figure 3(c, d) for Mamba) that is not provided by (B1')–(B2) alone.*

Remark A.5 (Honest scope of Theorem A.6). Theorem A.6 does not prove $\rho_s(I_a, I_\Delta) \geq 0$ on selective SSMs. It proves that the architectural mechanism forcing $\rho_s < 0$ in residual transformers (large δ from privileged input injection plus terminal-task alignment) is absent in selective SSMs. The empirical positive correlation reported in Table 11 is therefore consistent with the theory but is an additional empirical fact, not a proven consequence. The previous formulation overclaimed by asserting $\rho_s \rightarrow 1$ as $L \rightarrow \infty$; the present statement reflects what the assumptions actually deliver.

Remark A.6 (Architectures the theorem covers). The proof uses only (B1') and (B2). Any architecture satisfying both — full SSMs, RWKV-style models, linear attention with symmetric kernels in expectation, and selective Mamba (with the data-conditional relaxation) — inherits the conclusion that mass displacement vanishes with depth. Architectures satisfying only (B1') but not (B2) (e.g. a residual model with symmetric path-product but f_0 -privileged input injection) inherit only a partial vanishing of δ , consistent with $\rho_s \approx 0$ rather than $\rho_s > 0$.

A.5 Corollary: Scale-Induced Collapse, Not Reversal

The two largest transformers in our experiments deviate from the Theorem A.3 regime: their placement ranking alignment collapses toward zero rather than remaining strongly negative (Section 5.4). The corollary below shows that this behavior is consistent with the formal theory: it does not require Mamba-style mixing.

Corollary A.8 (Scale-induced collapse). *Suppose Assumption A.1(A1)–(A2) hold but (A3) is replaced by a bimodal mass split for I_Δ : $\tilde{I}_\Delta(\bar{E}) = \tau_{\text{late}}$ and*

$\tilde{I}_\Delta(E) = \tau_{\text{early}}$ with $\tau_{\text{late}} + \tau_{\text{early}} \leq 1$. The ablation mass-displacement bound (12) is unchanged. The mass-disagreement coefficient becomes

$$\delta = \tilde{I}_a(E) + \tau_{\text{late}} - 1 \geq \frac{\kappa}{\kappa + L(1 + \beta)^L} + \tau_{\text{late}} - 1. \quad (17)$$

In particular, δ vanishes (and the Lemma A.4-induced bound on ρ_s becomes vacuous) at the threshold

$$\tau_{\text{late}}^* = 1 - \frac{\kappa}{\kappa + L(1 + \beta)^L}. \quad (18)$$

Below this threshold, the architectural mechanism no longer forces $\rho_s < 0$; the empirical $|\rho_s|$ is then expected to scale linearly with $\tau_{\text{late}} - \tau_{\text{late}}^*$.

Proof. The bound on $\tilde{I}_a(E)$ is unchanged because (A1)-(A2) are unchanged. By assumption $\tilde{I}_\Delta(\bar{E}) = \tau_{\text{late}}$. Substituting into the definition $\delta = \tilde{I}_a(E) + \tilde{I}_\Delta(\bar{E}) - 1$ gives (17). The threshold (18) is the value of τ_{late} at which the right-hand side of (17) equals zero. $\square$

Remark A.7. The corollary makes a sharp empirical claim: scaling depth on a residual transformer can drive $|\rho_s|$ toward 0 but *cannot* flip its sign to positive without also weakening the input-injection asymmetry (A2). The ablation-side mass bound (12) is unchanged at scale, so a sign reversal would require either a structural change (hybrid SSM blocks) or a parameterization change (input injection at every layer). The Qwen3-32B observation (large layer-0 ablation share, near-zero ρ_s , rather than $\rho_s > 0$) is consistent with this prediction: only the LoRA-delta profile reorganized, not the ablation profile.

A.6 Falsifiable predictions

The formal theory generates five concrete, falsifiable predictions. (P1)-(P3) are partially confirmed in the body of the paper; (P4)-(P5) are open.

(P1) Leave-out-L0 in transformers. Removing layer 0 from both I_a and I_Δ in a residual transformer should reduce $|\rho_s|$ (by reducing $\tilde{I}_a(E)$ in the mass-displacement bound (12)) but should not flip its sign, because $\tilde{I}_a(E)$ remains $> 1/2$ even after removing the layer-0 contribution as long as input-side mass is concentrated more broadly near the bottleneck. Confirmed in Section 5.3 and Appendix E.

(P2) Architecture transfer. Any architecture satisfying Assumption A.2(B1')-(B2) — full SSMs, RWKV, linear attention with symmetric kernels in expectation — should exhibit $\delta \rightarrow 0$ as $L/K \rightarrow \infty$ and therefore lack the architectural mechanism forcing $\rho_s < 0$. Confirmed for Mamba on every paired checkpoint.

(P3) LoRA-target alignment. Choosing LoRA target modules whose subspace $\mathcal{S}_l$ is aligned with the dominant Hessian eigenbasis of θ_l should reduce the curvature factor in Equation (4), decreasing τ in Assumption A.1(A3) and weakening the late-half mass concentration of I_Δ . The submodule-matched control (Appendix E.2) is a partial realization; a full test requires varying the LoRA-target choice while holding everything else fixed.

(P4) Threshold predicts magnitude. Under Corollary A.8, $|\rho_s|$ should vanish at τ_{late}^* (Eq. (18)) and grow linearly with the gap $\tau_{\text{late}} - \tau_{\text{late}}^*$ above it. This is testable by computing the late-half LoRA-delta mass on each transformer checkpoint and regressing $|\rho_s|$ against the predicted threshold.

(P5) Per-layer-input transformer. If a residual transformer is modified to inject the embedding at every layer (so (A2) fails because f_0 is no longer the unique initialization of the residual stream), the mass-displacement bound on $\tilde{I}_a(E)$ degrades to the Theorem A.6 regime, $\delta \rightarrow 0$, and ρ_s should no longer be forced negative. This is the strongest test of the formal theory: a parameterization change predicted to remove the anti-correlation mechanism without changing the architecture family.

P5 is a falsification target. If a per-layer-input transformer still shows $\rho_s < 0$, then Assumption (A2) is not the load-bearing structural property the theory claims it is, and the mechanism would have to be located elsewhere (e.g. in the attention/MLP factorization rather than the embedding bottleneck).

A.7 Connection to prior theoretical work

The general claim that gradient, ablation, and update-magnitude attributions probe distinct functionals is established in the attribution literature: Sundararajan et al. (2017) prove that no method satisfies the *Sensitivity, Implementation Invariance*,

and *Completeness* axioms simultaneously; Ancona et al. (2018) unify gradient-based methods as varying-fidelity linearizations of true ablation; Adedbayo et al. (2020) show empirically that gradient-saliency maps survive parameter randomizations that destroy the network’s behavior. Our contribution is to make the corresponding statement precise at the *layer level*, derive the specific transformer pattern from residual-stream gradient flow under explicit assumptions, and quantify the practical consequence as a mass-displacement bound (rigorous from Assumption A.1 alone) coupled with a separate rank-correlation lemma (Lemma A.4, conditional on profile-shape monotonicity).

The closest prior decomposition is the disagreement-attribution framework of Krishna et al. (2024): observe disagreement, neither derives a quantitative bound from architectural structure. Theorem A.3 provides such a bound, and Theorem A.6 establishes its architectural specificity. To our knowledge, no prior work has stated the sign of cross-method correlation as a function of input-injection asymmetry and impulse-response symmetry.

B Experimental Details

B.1 Background

Low-Rank Adaptation (LoRA) (Hu et al., 2021) fine-tunes a pretrained model by inserting trainable low-rank updates $\Delta W = BA$ into selected weight matrices ($B \in \mathbb{R}^{d \times r}$, $A \in \mathbb{R}^{r \times k}$, $r \ll \min(d, k)$), keeping the base parameters frozen. Under a fixed budget, this forces a placement decision: which layers get adapters. *Importance-guided* placement (Zhang et al., 2023; Qing et al., 2024; Xu et al., 2026; Zhao et al., 2026; Hayou et al., 2026) replaces uniform allocation by estimating which layers matter most for a target task. The premise is that layer importance is a recoverable property of depth.

We write the importance of layer l for task t under method m as $I_m(l, t) \in \mathbb{R}_{\geq 0}$. The key observation is that “matters” is not uniquely defined. The five estimators we study fall into three categories: *ablation* measures functional necessity (which computations the model already depends on, scored by the loss increase from disabling a layer-local computation); *LoRA delta attribution* measures adaptation pressure (where the optimizer writes task-specific updates during fine-tuning, scored by the merged update magnitude); and three sensi-

tivity proxies (gradient attribution, resnorm (Xu et al., 2026), activation norm) are training-free signals serving as diagnostic baselines. Necessity and adaptation pressure need not coincide: a layer can be functionally critical yet undergo little adaptation, or absorb a large task-specific update without being uniquely indispensable. Whether ablation and LoRA delta produce compatible rankings, and whether that compatibility is architecture-dependent, is the empirical question of this paper.

B.2 Models

Table 1: Models used in this study. L denotes the number of layers (transformer blocks or SSM blocks). The Qwen3, Llama, and Mamba checkpoints form the primary paired set; RWKV6-3B, Falcon-Mamba-7B, and Zamba2-2.7B are validation architectures. Zamba2-2.7B’s 54 layers comprise 45 Mamba blocks and 9 shared-attention hybrid blocks at positions 6, 12, ..., 51.

Family	Model	Params	L
Qwen3 (Yang et al., 2025)	Qwen3-0.6B	0.6B	28
	Qwen3-1.7B	1.7B	28
	Qwen3-4B	4B	36
	Qwen3-8B	8B	36
	Qwen3-14B	14B	40
	Qwen3-32B	32B	64
Llama (Grattafiori et al., 2024)	Llama-3.2-1B	1B	16
	Llama-3.2-3B	3B	28
	Llama-3.1-8B	8B	32
	Llama-3.1-70B	70B	80
Mamba (Gu and Dao, 2024)	Mamba-130m	130M	24
	Mamba-370m	370M	48
	Mamba-790m	790M	48
	Mamba-2.8B	2.8B	64
Other / hybrid	RWKV6-3B (Peng et al., 2023)	3B	32
	Falcon-Mamba-7B (Zuo et al., 2024)	7B	64
	Zamba2-2.7B (Glorioso et al., 2024)	2.7B	54

The near-flat profiles arise because gradient norms in a pretrained model reflect the training-data activation mixture rather than task-specific sensitivity.

B.3 Discriminability across models and methods

B.4 Necessity–Plasticity projection of additional estimators

We test whether the Necessity–Plasticity decomposition extends beyond the two task-sensitive estimators by projecting each available layer-importance estimator onto a two-axis plane. The Plasticity loading ρ_P is the mean Spearman correlation between the estimator’s per-layer scores and LoRA-delta scores. The Necessity loading ρ_N is the mean Spearman correlation between the estimator’s per-layer scores and ablation scores.

On transformers, four of five additional estimators load clearly on one side: Gradient ($\rho_P =$

Table 2: Task-native MCQ accuracy audit for causal-validation rows. Each entry averages top- k and bottom- k accuracy over the available MCQ tasks: MMLU, ARC-Challenge, GPQA, PIQA, and WinoGrande. When multiple runs are available for the same model and method, we average them. Δ is top- k minus bottom- k . Generation-style tasks are excluded because their main diagnostic validation uses perplexity rather than task-native pass@1/F1/exact-match metrics.

Model	Method	Top- k	Bottom- k	Δ
Qwen3-8B	LoRA delta	0.500	0.522	-0.022
Qwen3-8B	Ablation	0.524	0.593	-0.069
Qwen3-8B	Gradient	0.514	0.525	-0.011
Qwen3-8B	Resnorm	0.566	0.514	+0.052
Qwen3-8B	ShapLoRA	0.491	0.546	-0.055
Qwen3-8B	TELL-TALE	0.498	0.530	-0.031
Llama-3.1-8B	LoRA delta	0.471	0.549	-0.078
Llama-3.1-8B	Ablation	0.555	0.517	+0.038
Llama-3.1-8B	Gradient	0.559	0.513	+0.047
Llama-3.1-8B	Resnorm	0.541	0.530	+0.011
Llama-3.1-8B	ShapLoRA	0.572	0.530	+0.042
Llama-3.1-8B	TELL-TALE	0.517	0.555	-0.038
Qwen3-14B	LoRA delta	0.544	0.538	+0.006
Qwen3-14B	Random	0.543	0.522	+0.021
Mamba-790m	LoRA delta	0.451	0.466	-0.015
Mamba-790m	Ablation	0.617	0.672	-0.055
Mamba-790m	Resnorm	0.555	0.549	+0.006
Mamba-790m	ShapLoRA	0.444	0.474	-0.030
Mamba-790m	TELL-TALE	0.452	0.472	-0.020

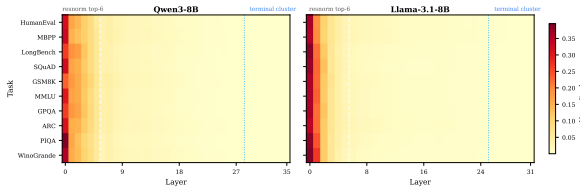


Figure 8: Resnorm profiles for Qwen3-8B and Llama-3.1-8B. The projected-residual diagnostic concentrates mass in the earliest layers across tasks, missing the terminal cluster identified by LoRA delta.

-0.66 , $\rho_N = +0.47$) and Resnorm ($\rho_P = -0.89$, $\rho_N = +0.46$) project onto Necessity; Activation norm ($\rho_P = +0.84$, $\rho_N = -0.35$) and TELL-TALE ($\rho_P = +0.65$, $\rho_N = -0.32$) project onto Plasticity; ShapLoRA lies near the boundary ($\rho_P = +0.22$, $\rho_N = -0.08$). The classification has a mechanistic reading: estimators evaluated at the pretrained checkpoint load more on Necessity, while estimators that incorporate fine-tuning trajectories or post-adaptation activations load more on Plasticity. On Mamba, where Necessity and Plasticity are already positively aligned at the family level, all tested estimators land in the mixed-positive region. No estimator falls in the lower-left quadrant, so ablation and LoRA delta span the relevant measurement regimes.

Table 3: Minimum pairwise cosine similarity ($d_{\min}$) across models and methods. Lower values indicate stronger task discrimination. Bold indicates the most discriminative method per model. “—” indicates the method was not run for that model.

Model	L	Gradient	Ablation	LoRA delta
Qwen3-0.6B	28	0.980	0.999	0.981
Qwen3-1.7B	28	0.959	1.000	0.962
Qwen3-4B	36	0.977	0.923	0.952
Qwen3-8B	36	0.979	0.949	0.959
Qwen3-14B	40	0.980	0.950	0.948
Qwen3-32B	64	0.977	0.952	0.941
Llama-3.2-1B	16	0.989	0.800	0.962
Llama-3.2-3B	28	0.977	0.928	0.967
Llama-3.1-8B	32	0.962	0.967	0.980
Llama-3.1-70B	80	—	0.961	0.971
Mamba-130m	24	0.998	0.953	0.942
Mamba-370m	48	—	0.902	0.921
Mamba-790m	48	—	0.798	0.823
Mamba-2.8B	64	—	0.843	0.861

C Multi-Seed and Budget Robustness

C.1 Multi-seed causal validation

We repeat all causal-validation runs for Qwen3-8B and Llama-3.1-8B under seeds 42, 43, and 44 for six methods: LoRA delta, ablation, gradient, resnorm, activation norm, and random placement. This section reports aggregate and per-task robustness results.

Metrics. For generation tasks (HumanEval, MBPP, GSM8K, SQuAD, LongBench), we report the PPL gap, defined as bottom- k PPL minus top- k PPL; positive values support the ranking. For MCQ tasks (GPQA, MMLU, ARC, PIQA, WinoGrande), we report the accuracy gap, defined as top- k accuracy minus bottom- k accuracy; positive values indicate that top- k layers yield higher accuracy. The binary hypothesis_supported flag is derived from the appropriate metric for each task type.

Summary. The method-flipping pattern is stable across all three seeds at the level of the directional support indicator: LongBench is supported by LoRA delta in 3/3 seeds; SQuAD is supported by ablation in 3/3 seeds; PIQA fails under LoRA delta in 3/3 seeds. Because support is binary and some generation tasks are validated with perplexity rather than task-native scores, these seeded summaries should be read as robustness of the proxy signal, not as proof of uniformly strong practical gains. Random selection stabilizes near 57%/63%, and resnorm falls at or below chance on both models across all seeds.

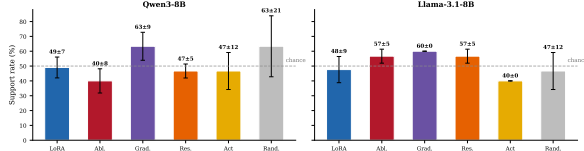


Figure 9: Mean directional causal-validation support rate $\pm$ std across three seeds for Qwen3-8B and Llama-3.1-8B. The dashed line marks 50%. Error bars show std across seeds. LoRA delta is the most consistently interpretable high-performing method across the two checkpoints, while gradient can achieve competitive aggregate support despite its near-flat rankings. Resnorm falls at or below random on both models.

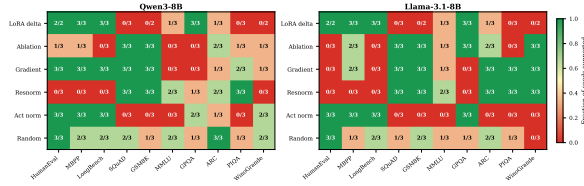


Figure 10: Fraction of seeds where the causal hypothesis is directionally supported, per task and method. Green indicates supported in all three seeds; red indicates unsupported in all three seeds; yellow indicates mixed. Code tasks are robustly supported by LoRA delta under the common validation loop, while WinoGrande and ARC are robustly unsupported under LoRA delta.

C.2 Seed coverage

C.3 Causal-validation margin magnitudes

The main text uses top- k versus bottom- k support labels to test whether an importance ranking has directional causal consequences under selective LoRA placement. These labels record the sign of the comparison, not the magnitude of the downstream change. We therefore report the corresponding metric gaps here. For generation-style tasks evaluated with perplexity, the gap is $\text{PPL}_{\text{bottom}} - \text{PPL}_{\text{top}}$, so positive values favor the top- k layers. For multiple-choice tasks evaluated with accuracy, the gap is $\text{Acc}_{\text{top}} - \text{Acc}_{\text{bottom}}$.

Figure 11 reports per-task gaps with seed error bars for Qwen3-8B and Llama-3.1-8B. Tables 7, 8, and 5 report primary-metric gaps for competitor placement methods, the Qwen3-14B and Mamba-790m extension runs, and the k -sweep respectively.

The margin results show three regimes. First, a small number of task-model-method triples have substantively large gaps, including Qwen3-1.7B on SQuAD under ablation, Qwen3-1.7B on WinoGrande where bottom- k ablation layers win by a large margin, and Mamba-790m on SQuAD un-

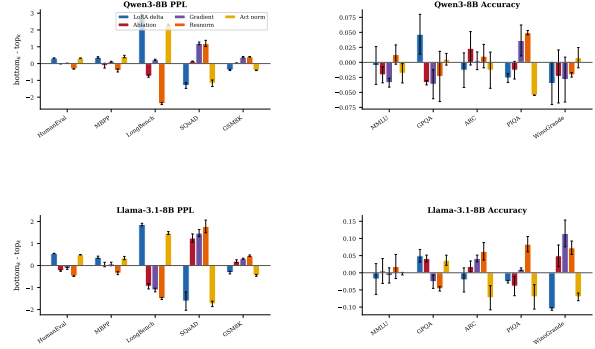


Figure 11: Metric gaps across seeds. Left column: PPL gap (bottom $_k$ minus top $_k$) for generation tasks. Right column: accuracy gap (top $_k$ minus bottom $_k$) for MCQ tasks. Positive values support the hypothesis. Error bars show std across three seeds. These magnitudes should be inspected directly rather than inferred from the binary support label alone.

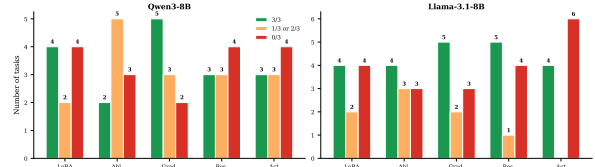


Figure 12: Cross-seed stability of support decisions. Green indicates supported in all three seeds; orange indicates mixed support; red indicates unsupported in all three seeds. LoRA delta shows the highest fraction of fully stable supported tasks for code and long-context. Commonsense tasks are stably unsupported under LoRA delta across seeds.

der LoRA delta. Second, most supported pairs are small in absolute magnitude, with typical primary-metric gaps in the 0.01–0.05 range. Third, random placement, competitor methods, and several larger-scale extension runs remain close to zero on average. Thus, the causal-validation results should not be read as evidence that any single estimator provides a large universal downstream gain. They instead show that the direction of selective-placement validation is method-relative, even when the absolute margins are modest.

C.4 Sensitivity of causal validation to the budget k

The main protocol fixes $k = \lceil L/6 \rceil$. To check how much of the mixed-to-negative LoRA-delta verdict depends on that particular budget, we run an additional k -sweep for six transformer checkpoints across both families. For each model, we evaluate four values of k spanning roughly $k/L \in [0.08, 0.31]$ and report directional support together

Table 4: Per-checkpoint multi-seed coverage of the causal-validation runs. Brace-enclosed lists are the seed indices with completed result files. “LoRA- δ support per seed” shows the number of directionally supported tasks at each completed seed.

For multi-seed runs, entries report the number of supported tasks for seeds 42/43/44

Model	Family	LoRA- δ support across seeds
Qwen3-0.6B	Tx	3/3/2 of 10
Qwen3-1.7B	Tx	5/5/5 of 10
Qwen3-4B	Tx	4/3/6 of 10
Qwen3-8B	Tx	4/5/4 of 7
Qwen3-14B	Tx	7/7/8 of 10
Qwen3-32B	Tx	7/8/9 of 10
Llama-3.2-1B	Tx	6/4/3 of 10
Llama-3.2-3B	Tx	2/5/6 of 7
Llama-3.1-8B	Tx	3/6/4 of 7
Llama-3.1-70B	Tx	7 of 10
Mamba-130m	SSM	8/6/8 of 10
Mamba-370m	SSM	8/7/8 of 10
Mamba-790m	SSM	6/6/7 of 10
Mamba-2.8b	SSM	7/7/7 of 10

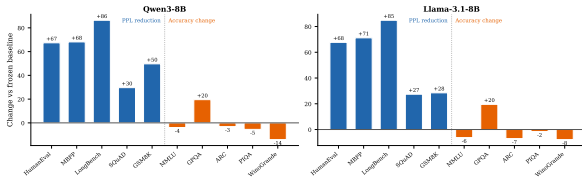


Figure 13: Performance change under uniform LoRA relative to the frozen model. Uniform LoRA substantially improves generation perplexity, but does not uniformly improve multiple-choice accuracy.

with the mean top_k -minus-bottom $_k$ primary-metric gap, using the same task-native metrics and the same LoRA rank and training budget as the main experiments.

The qualitative verdict is insensitive to the budget choice: the $k = \lceil L/6 \rceil$ row sits inside the envelope swept by $k \in [2, 8]$ at each checkpoint, and neither family benefits consistently from a different k . Across all twelve Llama (model, k) points, mean support is 0.23 and the mean primary gap is -0.025 ; across the twelve Qwen points, mean support is 0.29 and the mean primary gap is $+0.005$. The method-relative disagreement reported in the main text is therefore not an artifact of the fixed budget. The caveat is that this sweep was not run on Mamba checkpoints or on the largest transformers, and it sweeps only LoRA delta rather than all estimators.

C.5 Task-A adaptation quality: top_k versus bottom- k

A potential confound in the continual-learning protocol is that bottom- k layers (least plastic) may simply learn Task A less thoroughly than top- k

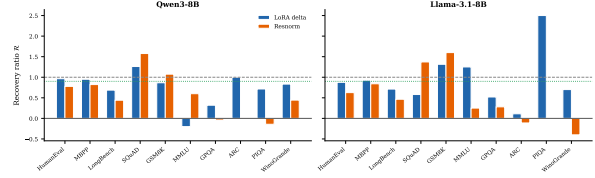


Figure 14: Recovery ratio $\mathcal{R}$ for top- k placement under LoRA delta and resnorm. LoRA delta recovers most of the full-LoRA improvement on code generation, while resnorm is inconsistent across tasks.

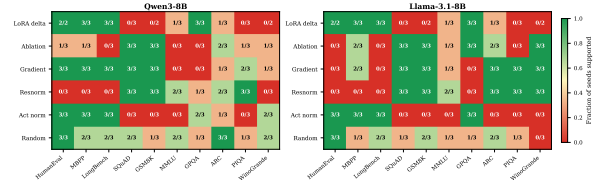


Figure 15: Per-task causal-validation support across methods, shown as the fraction of seeds in which each method supports the hypothesis. Code tasks are the most consistently recoverable; commonsense tasks are the least.

layers, mechanically producing a lower forgetting ratio without any genuine bottleneck effect. Table 6 addresses this directly by reporting the post-Task-A perplexity ($\text{PPL}_{\text{before}}$, measured after Task-A fine-tuning and before Task-B training) for both placement strategies across all models, together with the resulting forgetting ratios.

Two patterns emerge. First, in **transformers**, top- k does achieve modestly better Task-A adaptation: averaged across all transformer checkpoints and pairs ($n=75$), $\text{PPL}_{\text{before}}$ is 1.71 for top- k versus 2.00 for bottom- k — a 17% gap, meaning the least plastic layers absorb Task-A somewhat less. However, the forgetting ratio gap is substantially larger: 3.24 for top- k versus 2.33 for bottom- k — a 39% difference in the direction opposite to what a pure “learned less, forgets less” account would predict. A confounder that accounts for a 17% advantage in starting perplexity cannot explain a 39% disadvantage in forgetting ratio; the bottleneck effect is not an artifact of differential Task-A learning.

Second, in **SSMs**, top- k and bottom- k $\text{PPL}_{\text{before}}$ are virtually identical (6.53 vs. 6.73, a 3% gap, $n=54$), and the forgetting ratios are likewise statistically indistinguishable (2.11 vs. 2.00). This confirms that the null forgetting result in SSMs is not driven by top- k and bottom- k learning at different rates.

Table 5: k -sweep causal validation on transformer checkpoints with LoRA-delta top- k . Each row is one $(model, k)$ configuration; all runs use the ten-task suite. Support is the fraction of tasks on which top- k outperforms bottom- k on the task-native primary metric. Mean primary gap is averaged across tasks; positive means top- k is better.

Model	Family	k	k/L	Support	Mean primary gap
Llama-3.2-1B	Llama	2	0.125	5/10	+0.002
Llama-3.2-1B	Llama	3	0.188	3/10	+0.005
Llama-3.2-1B	Llama	4	0.250	3/10	-0.006
Llama-3.2-1B	Llama	5	0.312	5/10	+0.002
Llama-3.2-3B	Llama	3	0.107	2/10	-0.031
Llama-3.2-3B	Llama	4	0.143	1/10	-0.033
Llama-3.2-3B	Llama	5	0.179	1/10	-0.040
Llama-3.2-3B	Llama	7	0.250	2/10	-0.026
Llama-3.1-8B	Llama	3	0.094	2/10	-0.040
Llama-3.1-8B	Llama	5	0.156	2/10	-0.049
Llama-3.1-8B	Llama	6	0.188	1/10	-0.055
Llama-3.1-8B	Llama	8	0.250	1/10	-0.027
<hr/>					
Qwen3-1.7B	Qwen	3	0.107	4/10	+0.010
Qwen3-1.7B	Qwen	4	0.143	5/10	+0.012
Qwen3-1.7B	Qwen	5	0.179	3/10	+0.002
Qwen3-1.7B	Qwen	7	0.250	4/10	+0.008
Qwen3-4B	Qwen	3	0.083	2/10	+0.021
Qwen3-4B	Qwen	5	0.139	3/10	+0.025
Qwen3-4B	Qwen	6	0.167	3/10	+0.021
Qwen3-4B	Qwen	8	0.222	3/10	+0.015
Qwen3-8B	Qwen	3	0.083	1/10	-0.019
Qwen3-8B	Qwen	5	0.139	1/10	-0.011
Qwen3-8B	Qwen	6	0.167	4/10	-0.005
Qwen3-8B	Qwen	8	0.222	2/10	-0.015
<hr/>					
Llama family	—	all	—	$\bar{s} = 0.23$	$\bar{g} = -0.025$
Qwen family	—	all	—	$\bar{s} = 0.29$	$\bar{g} = +0.005$

D Estimator and Architecture Extensions

D.1 Head-to-head comparison with published placement methods

We re-implement two competitor placement methods—ShapLoRA (Zhao et al., 2026) and TELL-TALE (Naim et al., 2026)—and run them through the same top- k versus bottom- k causal-validation protocol on Qwen3-8B, Llama-3.1-8B, and Mamba-790m at seed 42. This comparison is not intended as a refutation of either method in its original setting; the goal is to test whether the same matched diagnostic protocol produces a single estimator that dominates across architectures.

Table 7 reports directional support rates restricted to the task list common to every method on each model: a seven-task subset for the two transformer checkpoints, and a nine-task subset for Mamba-790m. On both transformer checkpoints, neither competitor beats random under the matched protocol. The same pattern holds on Mamba-790m, where both competitors land below LoRA delta. Mean primary-metric gaps are near zero or negative for both competitors on every tested model, indicating that the under-performance is directional rather than a thresholding artifact of the binary support metric.

Table 6: Task-A post-training perplexity (PPL_{before} , lower is better) and forgetting ratio ($PPL_{\text{after}}/PPL_{\text{before}}$, lower is better) for top- k vs. bottom- k placement across all CL models. Values are mean $\pm$ std across all task pairs and seeds. The “ratio” column shows $(\text{bottom-}k PPL_{\text{before}}) / (\text{top-}k PPL_{\text{before}})$; values above 1.0 indicate that top- k achieves better Task-A adaptation. Family summary rows pool all pairs within the family. RWKV6-3B omitted (numerical divergence in top- k runs).

Model	Family	PPL-before (Task-A fit)			Forgetting ratio	
		top- k	bottom- k	bot/top	top- k	bottom- k
Llama-3.2-3B	Transformer	2.01 $\pm$ 0.73	2.22 $\pm$ 0.57	1.11	2.67 $\pm$ 1.66	2.19 $\pm$ 0.91
Llama-3.1-8B	Transformer	1.80 $\pm$ 0.69	2.05 $\pm$ 0.57	1.14	2.86 $\pm$ 1.62	2.57 $\pm$ 1.48
Qwen3-0.6B	Transformer	1.87 $\pm$ 0.68	2.24 $\pm$ 0.36	1.20	6.83 $\pm$ 4.28	2.67 $\pm$ 1.01
Qwen3-4B	Transformer	1.52 $\pm$ 0.38	1.98 $\pm$ 0.30	1.30	2.66 $\pm$ 1.35	2.22 $\pm$ 0.80
Qwen3-8B	Transformer	1.39 $\pm$ 0.32	1.55 $\pm$ 0.37	1.11	2.85 $\pm$ 1.51	2.18 $\pm$ 1.06
<hr/>						
Mamba-130M	SSM	10.45 $\pm$ 3.57	10.80 $\pm$ 3.81	1.03	1.72 $\pm$ 0.29	2.09 $\pm$ 0.75
Mamba-370M	SSM	8.33 $\pm$ 4.08	8.58 $\pm$ 4.17	1.03	2.93 $\pm$ 1.52	1.48 $\pm$ 0.23
Mamba-790M	SSM	7.05 $\pm$ 3.23	7.18 $\pm$ 3.37	1.02	1.94 $\pm$ 0.57	1.85 $\pm$ 0.56
Mamba-2.8B	SSM	4.78 $\pm$ 1.57	5.18 $\pm$ 1.54	1.08	2.68 $\pm$ 1.27	3.45 $\pm$ 2.64
Mamba2-1.3B	SSM	6.55 $\pm$ 2.05	6.42 $\pm$ 2.30	0.98	1.57 $\pm$ 0.50	1.87 $\pm$ 0.49
Falcon-Mamba-7B	SSM	2.01 $\pm$ 0.43	2.21 $\pm$ 0.27	1.10	1.81 $\pm$ 0.62	1.30 $\pm$ 0.19
<hr/>						
Zamba2-2.7B	Hybrid	2.37 $\pm$ 0.78	2.74 $\pm$ 1.33	1.15	1.28 $\pm$ 0.16	1.43 $\pm$ 0.41
<hr/>						
Transformer	— ($n=75$)	1.71 $\pm$ 0.61	2.00 $\pm$ 0.51	1.17	3.24 $\pm$ 2.40	2.33 $\pm$ 1.05
SSM	— ($n=54$)	6.53 $\pm$ 3.77	6.73 $\pm$ 3.88	1.03	2.11 $\pm$ 1.01	2.00 $\pm$ 1.31
Hybrid	— ($n=9$)	2.37 $\pm$ 0.78	2.74 $\pm$ 1.33	1.15	1.28 $\pm$ 0.16	1.43 $\pm$ 0.41

D.2 Scale and architecture extension: Qwen3-14B and Mamba-790m

The Qwen3-8B and Llama-3.1-8B seeded study leaves two natural questions open: whether the scale-dependent weakening of LoRA delta persists at larger transformer sizes, and whether the transformer-SSM contrast survives a seeded causal-validation setup. To address both, we rerun the causal-validation pipeline on Qwen3-14B and Mamba-790m at seeds 43 and 44, with the same $k = 6$, LoRA rank 8, and 300 training steps as the 8B study. Each model is run under three placement regimes: LoRA-delta top- k , random top- k , and uniform LoRA over all layers. For Mamba-790m, we report aggregates on the nine non-LongBench tasks because the A1 task list and the Mamba-790m importance scores use different LongBench variants.

Table 8 summarizes directional support rates and mean primary-metric gaps. On Mamba-790m, LoRA delta yields a positive mean top $_k$ -minus-bottom $_k$ primary-metric gap at both seeds, while random top- k is near zero or negative. On Qwen3-14B, the picture is inverted: random top- k matches or exceeds LoRA delta both on support rate and on mean top-minus-bottom gap. This is consistent with the Qwen3-8B to Qwen3-32B bimodality reported in Figure 21: in the transformer family, LoRA delta’s prescriptive content weakens as scale grows, while in the SSM family it remains informative under the same validation recipe.

Table 7: Causal-validation comparison against published placement methods at seed 42. Each block restricts to the task list common to every method listed for that model. Support is the fraction of tasks on which top- k outperforms bottom- k under the loss-aligned validation metric. Mean PPL gap is bottom- k minus top- k on perplexity-validated tasks; mean primary gap is top- k minus bottom- k on task-native primary metrics. Empty primary-gap entries indicate runs whose logs predate the task-native evaluation path. LoRA-delta primary-gap entries are taken from the task-native audit and may use a different task subset.

Method	Support	Mean PPL gap	Mean primary gap
<i>Qwen3-8B</i> ($N_{\text{tasks}} = 7$)			
LoRA delta (Hu et al., 2021)	4/7 (57%)	-1.359	+0.001 [†]
Ablation	3/7 (43%)	+0.771	—
Gradient	5/7 (71%)	+1.604	—
Resnorm (Xu et al., 2026)	3/7 (43%)	+1.320	—
Random	4/7 (57%)	-0.106	—
ShapLoRA (Zhao et al., 2026)	3/7 (43%)	+0.126	-0.031
TELL-TALE (Naim et al., 2026)	2/7 (29%)	-1.078	-0.010
<i>Llama-3.1-8B</i> ($N_{\text{tasks}} = 7$)			
LoRA delta (Hu et al., 2021)	3/7 (43%)	-1.471	-0.044 [‡]
Ablation	4/7 (57%)	+1.270	—
Gradient	4/7 (57%)	+1.296	—
Resnorm (Xu et al., 2026)	4/7 (57%)	+1.123	—
Random	4/7 (57%)	-0.750	—
ShapLoRA (Zhao et al., 2026)	3/7 (43%)	+0.677	+0.048
TELL-TALE (Naim et al., 2026)	1/7 (14%)	-0.761	-0.051
<i>Mamba-790m</i> ($N_{\text{tasks}} = 9$)			
LoRA delta (Hu et al., 2021)	5/9 (56%)	+0.502	-0.026 [§]
ShapLoRA (Zhao et al., 2026)	2/9 (22%)	+0.164	-0.008
TELL-TALE (Naim et al., 2026)	2/9 (22%)	+0.148	-0.012

D.3 Mamba scale sweep

Tables 9 and 10 report causal validation for all four Mamba checkpoints. Under ablation, top- k layers follow the boundary pattern at all scales. Under LoRA delta, top- k layers shift to mid-network positions and support rates remain high, including tasks such as WinoGrande and PIQA that fail systematically in transformers under LoRA delta.

E Alternative-Mechanism Controls

E.1 The sign flip is not a layer-0 artifact

The L0 control rules out the simplest explanation for the transformer-Mamba split. Transformers place much more ablation mass on layer 0 than Mambas, so the control is necessary. However, removing layer 0 shifts transformer correlations toward zero without changing the qualitative sign for transformers up to 14B. The observed anti-correlation is therefore not reducible to a single embedding-bottleneck layer.

E.2 The sign flip is not a submodule-scope artifact

In the main experiments, LoRA delta is computed on trainable LoRA target modules, whereas ablation removes a layer-local computation. This raises

Table 8: Scale/architecture extension: per-seed directional support rate and mean primary-metric gap for LoRA delta and random top- k , plus per-seed uniform-LoRA win rate against the no-adaptor baseline. Both checkpoints run with $k = 6$, LoRA rank 8, 300 training steps, at seeds 43 and 44.

Model	Method	Seed 43	Seed 44	Mean gap (s43, s44)	N_{tasks}
Qwen3-14B	LoRA delta	2/10 (20%)	4/10 (40%)	-0.017, -0.001	10
	Random	5/10 (50%)	5/10 (50%)	-0.000, +0.004	10
	Uniform LoRA	3/10 wins	2/10 wins	+0.062, +0.060	10
Mamba-790m [†]	LoRA delta	3/9 (33%)	3/9 (33%)	+0.006, +0.021	9
	Random	3/9 (33%)	2/9 (22%)	+0.001, -0.015	9
	Uniform LoRA	2/9 wins	3/9 wins	+0.034, +0.033	9

[†]Mamba-790m LongBench was not evaluated in the A1 extension: the updated A1 task list uses `longbench_multifieldqa_en` whereas the Mamba-790m LoRA-delta importance scores cover `longbench_single_doc`; regenerating Mamba importance against the new variant was out of scope. Rows therefore report the nine tasks common to both checkpoints.

A1 extension: LoRA delta vs. Random top- k on 9 common tasks (seeds 43, 44; LongBench excluded)

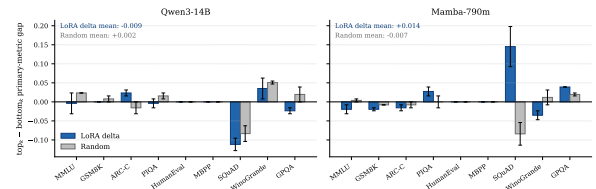


Figure 16: Per-task top- k minus bottom- k primary-metric gap on Qwen3-14B and Mamba-790m for LoRA delta and random top- k placement. Bars are seed-averaged means; error bars show the seed range over seeds 43 and 44. LongBench is excluded to keep the task set comparable across checkpoints.

a natural concern: perhaps the observed disagreement is caused by comparing different parts of the block rather than by comparing different notions of importance. To address this, we recompute placement ranking alignment under matched submodule scopes. For each matched setting, both estimators are restricted to the same part of the transformer block.

The cross-method anti-correlation is negative for every available submodule in Llama-3.1-8B: $\rho = -0.581$ for attention, -0.266 for MLP, and -0.298 for full-block comparisons. The corresponding top-6 layer overlaps are 15%, 17%, and 27%. For two random 6-of-32 selections, the expected overlap is $6 \times 6/32 = 1.125$ layers, or 18.75% of a top-6 set. Thus the attention and MLP overlaps are at or below random, whereas the full-block overlap remains far below within-LoRA overlaps. This rules out the possibility that the full-block anti-correlation is an artifact of mixing attention and MLP signals.

Within LoRA delta, submodules converge on

Table 9: Mamba causal validation under ablation across four model sizes. Top- k layers follow the boundary pattern at all scales.

Task	130m ($k=4$)	370m ($k=5$)	790m ($k=5$)	2.8b ($k=8$)
HumanEval	✓	✓	✓	✓
MBPP	✓	✓	✓	✓
LongBench	✓	✓	✓	✓
GSM8K	✓	×	✓	✓
SQuAD	✓	✓	✓	✓
MMLU	✓	✓	✓	✓
GPQA	✓	✓	✓	✓
ARC	✓	×	×	✓
PIQA	✓	×	×	×
WinoGrande	✓	✓	×	✓
Rate	10/10	7/10	7/10	9/10

Table 10: Mamba causal validation under LoRA delta across four model sizes. Mid-network layers are selected at all scales; support rates are 9–10/10.

Task	130m ($k=4$)	370m ($k=5$)	790m ($k=5$)	2.8b ($k=8$)
HumanEval	✓	✓	✓	✓
MBPP	✓	✓	✓	✓
LongBench	✓	✓	✓	✓
GSM8K	✓	✓	✓	✓
SQuAD	✓	✓	✓	✓
MMLU	✓	✓	×	×
GPQA	✓	✓	✓	✓
ARC	✓	✓	✓	✓
PIQA	✓	✓	✓	✓
WinoGrande	✓	✓	✓	✓
Rate	10/10	10/10	9/10	9/10

the same layers. The mean Spearman correlation between LD-attn and LD-MLP is $+0.839$ in Llama and $+0.659$ in Qwen, rising to $+0.883$ for LD-attn versus LD-full in Llama. Top-6 overlap ranges from 55–60%. This indicates that the terminal cluster is not a property of which projections are updated: adapting attention alone, MLP alone, or both simultaneously identifies the same final depth region.

Within ablation, attention and MLP criticality are more model-dependent. In Llama, Abl-attn versus Abl-MLP yields $\rho = +0.427$ and 67% top-6 overlap, suggesting moderate co-localization. In Qwen3-8B, the same comparison yields $\rho = -0.027$ and 27% overlap, indicating structurally distinct attention-critical and MLP-critical layer sets. Thus, if using ablation to guide submodule-specific placement, the submodule choice matters; if using LoRA delta, the choice is much less important.

E.3 The sign flip is not a LoRA artifact: adapter-free full fine-tuning control

Plasticity is defined in the main text through the magnitude of a LoRA update, which raises the possibility that $\mathcal{NPA}$ measures where LoRA places

Table 11: Necessity-Plasticity Alignment $\mathcal{NPA}(M)$ for the 14 primary paired pretrained checkpoints (Qwen3, Llama, Mamba). $\mathcal{NPA}(M)$ is the mean Spearman rank correlation between LoRA-delta and ablation importance vectors across $T = 10$ tasks. $\mathcal{NPA}^{\setminus L_0}(M)$ recomputes the same correlation with layer 0 excluded from both vectors. L0 share is the fraction of total ablation mass concentrated in the first layer. Family means in bold are arithmetic averages over the listed checkpoints. RWKV6-3B, Zamba2-2.7B, and Falcon-Mamba-7B are reported in Section 4.1 as validation architectures.

Model	Family	L	$C(M)$	$C(M) \setminus L_0$	L0 share
Qwen3-0.6B	Transformer	28	-0.25	-0.16	0.89
Qwen3-1.7B	Transformer	28	-0.47	-0.40	0.94
Qwen3-4B	Transformer	36	-0.23	-0.16	0.61
Qwen3-8B	Transformer	36	-0.30	-0.24	0.60
Qwen3-14B	Transformer	40	-0.06	-0.10	0.60
Qwen3-32B	Transformer	64	+0.07	+0.10	0.60
Llama-3.2-1B	Transformer	16	-0.74	-0.69	0.32
Llama-3.2-3B	Transformer	28	-0.48	-0.43	0.49
Llama-3.1-8B	Transformer	32	-0.62	-0.59	0.54
Llama-3.1-70B	Transformer	80	+0.01	+0.05	0.61
Mamba-130m	SSM	24	+0.22	+0.22	0.30
Mamba-370m	SSM	48	+0.22	+0.20	0.29
Mamba-790m	SSM	48	+0.25	+0.26	0.09
Mamba-2.8b	SSM	64	+0.28	+0.29	0.07
Transformer mean ($n=10$)			-0.31	-0.26	0.62
Transformer mean ($\leq 14B$, $n=8$)			-0.39	-0.35	0.62
Mamba mean ($n=4$)			+0.24	+0.24	0.19

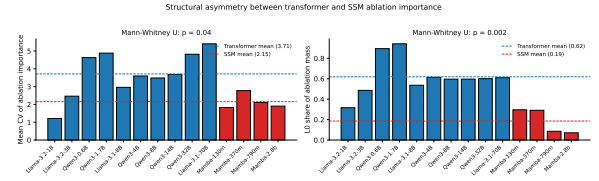


Figure 17: Per-checkpoint distribution of the structural asymmetry summarized by leave-out-L0 and L0-share controls. Dashed lines mark family means; two-sided Mann–Whitney U p -values compare family membership.

updates rather than where the architecture absorbs new information. We therefore rerun the Plasticity measurement without adapters: for each checkpoint we fine-tune *all* parameters on the same tasks, under the same data and budget, and recompute the per-layer update magnitude $\|\Delta W_l\|_F$ from the difference between the fine-tuned and pretrained weights. This removes every LoRA-specific degree of freedom at once: rank, scaling factor α , initialization, and the choice of target modules.

Table 14 reports the result on the 13 checkpoints we were able to fully retrain. Two quantities matter. First, $\rho(\Delta W^{\text{full}}, \Delta W^{\text{LoRA}})$ is positive on *every* checkpoint, so unconstrained fine-tuning and LoRA select the same layers; LoRA is not redirecting updates to an unrepresentative subset of depth. Second, $\rho(\Delta W^{\text{full}}, \text{ablation})$ — the adapter-free analogue of $\mathcal{NPA}$ — is negative for every evaluated transformer with an ablation baseline, and

Table 12: Submodule-matched placement ranking alignment. The Llama-3.1-8B full-block row is the headline control: both estimators are applied at the block level, yet the correlation remains negative. Within-LoRA-delta correlations between attention- and MLP-matched scorings are strongly positive, ruling out within-method noise.

Cross-method ρ	Attn-matched	MLP-matched	Full-block
Llama-3.1-8B	-0.58	-0.27	-0.30
Qwen3-8B	-0.07	+0.15	—

Table 13: Mean Spearman ρ between condition pairs for Llama-3.1-8B and Qwen3-8B. Cross-method comparisons are negative in Llama across all submodules; within-LoRA-delta comparisons are positive in both models; within-ablation comparisons reveal model-dependent divergence.

Comparison type	Model	Attn	MLP	Full
Abl vs LD (cross-method)	Llama-3.1-8B	-0.581	-0.266	-0.298
	Qwen3-8B	-0.066	+0.148	—
LD-attn vs LD-X (within LoRA delta)	Llama-3.1-8B	1.000	+0.839	+0.883
	Qwen3-8B	1.000	+0.659	—
Abl-attn vs Abl-X (within ablation)	Llama-3.1-8B	1.000	+0.427	+0.557
	Qwen3-8B	1.000	-0.027	+0.284

positive or near zero for every evaluated Mamba, RWKV, and hybrid checkpoint. The sign therefore matches the LoRA-based $\mathcal{NPA}$ of Table 11 in all 13 cases.

Correlation between the two adaptation procedures declines with model size (from 0.96 on Qwen3-0.6B to 0.39 on Qwen3-14B), which is expected: larger models have more directions in which a full-rank update can differ from a rank- r one. What does *not* decline is the agreement in sign between $\rho(\Delta W^{\text{full}}, \text{ablation})$ and $\mathcal{NPA}$, which is the quantity the paper’s claims rest on.

Update-to-weight normalization. A related concern is that $\|\Delta W_l\|_F$ rewards layers that simply have larger weight matrices. We therefore recompute the Plasticity ranking from the normalized quantity $\|\Delta W_l\|_F / \|W_l\|_F$, using the corresponding updated parameter set for each method. The qualitative conclusion is unchanged: LoRA-based $\mathcal{NPA}$ values are essentially stable under normalization, the evaluated transformers remain negative, the pure Mamba checkpoints and RWKV6-3B remain positive, and the already near-zero hybrid and boundary cases remain small in magnitude. Normalization thus rescales the Plasticity profile without moving the architecture split.

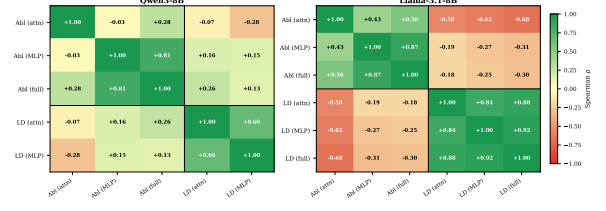


Figure 18: Full Spearman ρ matrix across matched-control conditions for Llama-3.1-8B and Qwen3-8B. The top-left block contains within-ablation correlations; the bottom-right block contains within-LoRA-delta correlations. The off-diagonal quadrants show the cross-method signal.

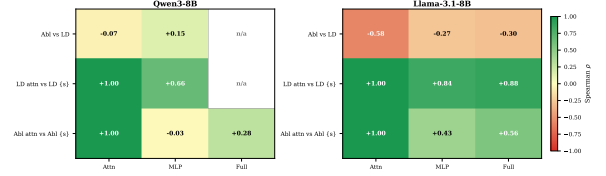


Figure 19: Summary Spearman ρ heatmap by comparison type and submodule. The cross-method row is negative for Llama across all submodules. The within-LoRA-delta row is positive for both models. The within-ablation row reveals that Qwen3-8B has near-zero attention-MLP correlation, indicating structurally separate attention-critical and MLP-critical layer sets.

E.4 Component-level analysis of a hybrid checkpoint

Zamba2-2.7B is the one checkpoint in our set that contains both block types, so it permits a within-model version of the architecture comparison. Its 54 layers comprise 45 Mamba blocks and 9 shared-attention hybrid blocks at positions 6, 12, ..., 51. We partition the layers by block type and compute each component’s share of three quantities: total LoRA update mass, total full-FT update mass, and total ablation Necessity mass.

Table 15 shows the dissociation reproduced inside a single model. The attention component is 16.7% of the layers but absorbs 36.3% of LoRA update mass and 31.6% of full-FT update mass — roughly twice its layer share under either adaptation procedure — while carrying only 10.4% of ablation Necessity mass, about $0.6\times$ its layer share. Necessity is instead concentrated in the Mamba blocks. Zamba2’s near-zero whole-model $\mathcal{NPA}$ of +0.05 is therefore not evidence that the effect is absent; it is the average of a transformer-like component and an SSM-like component with opposing profiles.

Because this rests on a single checkpoint with

Table 14: Adapter-free control. $\rho(\Delta W^{\text{full}}, \Delta W^{\text{LoRA}})$ is the Spearman correlation between full-parameter and LoRA per-layer update profiles. $\rho(\Delta W^{\text{full}}, \text{abl.})$ is the adapter-free analogue of $\mathcal{NP}\mathcal{A}$, computed against the same ablation-based Necessity rankings. The final column repeats the LoRA-based $\mathcal{NP}\mathcal{A}$ from Table 11 for comparison. Signs agree on all 13 checkpoints. Qwen3-32B, Llama-3.1-70B, Mamba-2.8B, and Falcon-Mamba-7B were not retrained under full FT for compute reasons.

Model	Family	$\rho(\text{full, LoRA})$	$\rho(\text{full, abl.})$	$\mathcal{NP}\mathcal{A}$
Qwen3-0.6B	Transformer	0.96	-0.27	-0.25
Qwen3-1.7B	Transformer	0.95	-0.45	-0.47
Qwen3-4B	Transformer	0.88	-0.21	-0.23
Qwen3-8B	Transformer	0.48	-0.25	-0.30
Qwen3-14B	Transformer	0.39	-0.23	-0.06
Llama-3.2-1B	Transformer	0.70	-0.76	-0.74
Llama-3.2-3B	Transformer	0.70	-0.64	-0.48
Llama-3.1-8B	Transformer	0.43	-0.42	-0.62
Mamba-130m	SSM	0.46	+0.11	+0.22
Mamba-370m	SSM	0.41	+0.35	+0.22
Mamba-790m	SSM	0.42	+0.41	+0.25
RWKV6-3B	SSM	0.75	+0.42	+0.54
Zamba2-2.7B	Hybrid	0.30	+0.05	+0.05

Table 15: Component-level decomposition of Zamba2-2.7B. Shares are percentages of the model total; $\pm$ values are standard deviations across tasks. The attention component takes roughly twice its layer share of adaptation mass under both LoRA and full FT, but only about $0.6\times$ its layer share of Necessity.

Component	Layers	LoRA ΔW	Full-FT ΔW	Necessity
Attention blocks (9)	16.7%	36.3 ± 0.6	31.6 ± 0.7	10.4 ± 2.6
Mamba blocks (45)	83.3%	63.7	68.4	89.6

one attention/SSM ratio, we treat statements about other ratios as extrapolation beyond the measured setting.

E.5 Proxy validation is conditional on the metric

The main causal-validation protocol uses a shared likelihood-based proxy so that top- k and bottom- k placement can be compared across heterogeneous tasks. This makes the comparison controlled, but it also means that the resulting directional labels should not be interpreted as universal claims about end-task performance.

Estimator choice reverses which tasks appear supported. LoRA-delta placement supports LongBench in 9/10 transformer checkpoints, whereas ablation supports it in 2/10. On SQuAD, the ordering flips: 3/10 under LoRA delta and 9/10 under ablation. These are the same models, training budget, and validation loop; only the importance estimator changes. Code-generation tasks favor LoRA delta under the proxy, while commonsense tasks show little structured support for either estimator.

Table 16: Task-native MCQ accuracy audit for causal-validation rows. Each entry averages top- k and bottom- k accuracy over the available MCQ tasks: MMLU, ARC-Challenge, GPQA, PIQA, and WinoGrande. When multiple runs are available for the same model and method, we average them. Δ is top- k minus bottom- k . Generation-style tasks are excluded because their main diagnostic validation uses perplexity rather than task-native pass@1/F1/exact-match metrics.

Model	Method	Top- k	Bottom- k	Δ
Qwen3-8B	LoRA delta	0.500	0.522	-0.022
Qwen3-8B	Ablation	0.524	0.593	-0.069
Qwen3-8B	Gradient	0.514	0.525	-0.011
Qwen3-8B	Resnorm	0.566	0.514	+0.052
Qwen3-8B	ShapLoRA	0.491	0.546	-0.055
Qwen3-8B	TELL-TALE	0.498	0.530	-0.031
Llama-3.1-8B	LoRA delta	0.471	0.549	-0.078
Llama-3.1-8B	Ablation	0.555	0.517	+0.038
Llama-3.1-8B	Gradient	0.559	0.513	+0.047
Llama-3.1-8B	Resnorm	0.541	0.530	+0.011
Llama-3.1-8B	ShapLoRA	0.572	0.530	+0.042
Llama-3.1-8B	TELL-TALE	0.517	0.555	-0.038
Qwen3-14B	LoRA delta	0.544	0.538	+0.006
Qwen3-14B	Random	0.543	0.522	+0.021
Mamba-790m	Ablation	0.617	0.672	-0.055
Mamba-790m	LoRA delta	0.451	0.466	-0.015
Mamba-790m	Random	0.455	0.449	+0.006
Mamba-790m	ShapLoRA	0.444	0.474	-0.030
Mamba-790m	TELL-TALE	0.452	0.472	-0.020

These patterns show that estimator choice changes the placement conclusion even when the training and validation loop are held fixed.

We then audit the proxy verdicts against task-native metrics, including pass@1, exact match, F1, and LongBench aggregate scores. The audit shows that the proxy metric is neither uniformly optimistic nor uniformly pessimistic. In some cases, the proxy reports a top- k win that does not appear under the native metric; in other cases, the native metric improves even when the proxy does not. Thus, proxy validation captures one operational notion of selective adaptation, but it is not interchangeable with task-native evaluation.

F Full Causal Validation Tables

Tables 17–25 present per-task causal-validation results for representative model–method pairs. Generation rows report perplexity; MCQ rows report accuracy where available. To keep the schema uniform, the metric columns below are labeled as Base, Top $_k$, and Bot $_k$. The final “Supp.” column is a directional indicator only. Rows marked with † are marginal near-ties and should not be treated as strong practical wins.

Table 17: Qwen3-8B causal validation under LoRA delta ($k = 6$ of 36 layers, seed 42). Generation rows report perplexity; MCQ rows report accuracy where available.

Task	Group	Top-6 layers	Base	Top _k	Bot _k	Supp.
HumanEval	Code	23,31–35	3.23	1.13	1.41	✓
MBPP	Code	28,29,32–35	5.88	2.07	2.49	✓
LongBench	Long-ctx	24,28,30–32,34	12.39	5.04	7.67	✓
GPQA	Science	28,31–35	6.65	3.57	3.79	✓
MMLU	Knowledge	30–35	10.79	9.97	6.15	×
GSM8K	Math	23,31–35	3.87	2.21	1.78	×
SQuAD	Retrieval	28,31–35	13.92	8.72	7.26	×
ARC	Commonsense	30–35	8.95	6.27	4.10	×
PIQA	Commonsense	30–35	12.45	9.80	4.46	×
WinoGrande	Commonsense	30–35	0.766	0.648	0.703	×

Table 18: Llama-3.1-8B causal validation under LoRA delta ($k = 6$ of 32 layers, seed 42). Generation rows report perplexity; MCQ rows report accuracy where available.

Task	Group	Top-6 layers	Base	Top _k	Bot _k	Supp.
HumanEval	Code	25–29,31	3.34	1.36	1.86	✓
MBPP	Code	24–29	5.93	2.01	2.33	✓
LongBench	Long-ctx	15,26–30	9.29	3.68	5.47	✓
GPQA	Science	16,17,19,26,29,30	5.28	3.09	3.40	✓
MMLU	Knowledge	15,17,23,25,28,29	8.27	10.31	6.03	×
GSM8K	Math	25–29,31	5.13	3.21	2.79	×
SQuAD	Retrieval	16,27–31	9.14	7.69	5.64	×
ARC	Commonsense	24–29	7.66	7.32	4.91	×
PIQA	Commonsense	24–29	11.56	9.25	5.28	×
WinoGrande	Commonsense	25–29,31	0.773	0.719	0.773	×

G Layer-Pattern Summaries

G.1 Per-model top-layer patterns

G.2 Structural layer-selection patterns by method

H Forgetting

Table 19: Mamba-790m causal validation under ablation ($k = 5$ of 48 layers). Boundary layers dominate top- k selection. Generation rows report perplexity; MCQ rows report accuracy where available.

Task	Group	Top-5 layers	Base	Top _k	Bot _k	Supp.
HumanEval	Code	0,1,34,46,47	4.59	3.95	4.18	✓
MBPP	Code	0,1,34,46,47	6.21	4.02	4.87	✓
GSM8K	Math	0,1,34,46,47	8.96	4.97	6.02	✓
SQuAD	Retrieval	0,1,29,46,47	9.14	5.31	6.78	✓
LongBench	Long-ctx	0,1,34,46,47	11.30	7.42	8.95	✓
MMLU	Knowledge	0,1,34,46,47	8.10	5.60	6.20	✓
GPQA	Science	0,1,34,46,47	6.20	4.10	4.58	✓
ARC	Commonsense	0,1,29,34,47	7.80	5.10	5.62	×
PIQA	Commonsense	0,1,29,46,47	10.40	7.80	8.20	×
WinoGrande	Commonsense	0,1,29,46,47	0.656	0.633	0.672	×

Table 20: Llama-3.2-3B causal validation under LoRA delta ($k = 5$ of 28 layers). Generation rows report perplexity; MCQ rows report accuracy where available.

Task	Group	Top-5 layers	Base	Top _k	Bot _k	Supp.
HumanEval	Code	19,20,23,26,27	3.54	1.75	2.16	✓
MBPP	Code	16,20,23,26,27	6.25	2.14	2.54	✓
LongBench	Long-ctx	16,23,25,26,27	10.80	6.42	7.30	✓
GPQA	Science	15,16,17,20,25	5.72	3.84	3.90	×
MMLU	Knowledge	14,20,25,26,27	9.00	7.71	6.64	×
SQuAD	Retrieval	15,19,22,24,27	12.42	9.61	8.64	×
ARC	Commonsense	20,23,24,25,27	8.14	5.35	4.96	×
PIQA	Commonsense	21,22,25,26,27	11.48	6.19	4.71	×
WinoGrande	Commonsense	19,20,23,26,27	0.766	0.680	0.750	×

Table 21: Qwen3-32B causal validation under LoRA delta ($k = 8$ of 64 layers). Generation rows report perplexity; MCQ rows report accuracy where available. The selected layers are bimodal, with an early cluster and a terminal cluster.

Task	Group	Top-8 layers (sample)	Base	Top _k	Bot _k	Supp.
HumanEval	Code	4,5,6,59,60,61,62,63	1.98	1.36	1.49	✓
MBPP	Code	53,54,55,59,60,61,62,63	4.37	2.03	2.48	✓
LongBench	Long-ctx	46,48,49,50,51,52,53,54	9.84	6.76	11.17	✓
GPQA	Science	2,4,5,6,51,52,61,62	5.16	3.54	3.88	✓
GSM8K	Math	4,5,6,16,17,60,61,62	2.25	1.38	1.45	✓
SQuAD	Retrieval	3,4,5,6,54,59,60,61	8.78	4.61	4.88	✓
MMLU	Knowledge	2,3,4,5,6,48,62,63	7.45	5.16	5.50	×
ARC	Commonsense	2,4,5,6,59,60,61,63	6.58	3.67	3.99	×
PIQA	Commonsense	4,5,6,59,60,61,62,63	7.67	4.10	4.17	×
WinoGrande	Commonsense	51,54,58,59,60,61,62,63	0.773	0.727	0.781	×

Table 22: Qwen3-32B causal validation under ablation ($k = 8$ of 64 layers). Generation rows report perplexity; MCQ rows report accuracy where available. WinoGrande is directionally supported, but the margin is effectively a tie.

Task	Group	Top-8 layers (sample)	Base	Top _k	Bot _k	Supp.
HumanEval	Code	0,1,6,45,46,47,48,63	1.98	1.16	1.19	✓
MBPP	Code	0,1,6,40,53,54,60,61	4.37	1.83	2.35	✓
LongBench	Long-ctx	0,1,4,5,6,53,60,61	9.84	6.76	7.30	✓
GPQA	Science	0,1,2,6,35,54,60,61	5.16	3.54	3.60	✓
SQuAD	Retrieval	0,1,6,38,48,52,60,61	8.78	4.61	4.73	✓
MMLU	Knowledge	0,1,2,6,35,53,62,63	7.45	5.16	5.26	✓
ARC	Commonsense	0,1,2,6,35,54,60,61	6.58	3.67	3.78	✓
WinoGrande	Commonsense	0,1,6,41,42,44,52,62	0.773	0.719	0.711	✓ [†]
GSM8K	Math	0,1,6,48,50,52,60,61	2.25	1.33	1.30	×
PIQA	Commonsense	0,6,41,44,52,54,60,63	7.67	4.10	4.04	×

Table 23: Qwen3-8B: top-5 layers by LoRA-delta attribution per task. Bold entries indicate layers outside the terminal cluster L33–L35.

Task	#1	#2	#3	#4	#5	Mean $\ \Delta\ $
MMLU	L34	L33	L30	L32	L35	3.07
GSM8K	L34	L35	L33	L32	L31	2.02
ARC	L34	L33	L35	L31	L32	2.38
PIQA	L34	L33	L32	L31	L35	2.37
HumanEval	L35	L34	L33	L25	L26	2.28
MBPP	L33	L34	L35	L32	L29	2.79
SQuAD	L34	L35	L33	L32	L31	2.25
WinoGrande	L32	L34	L33	L31	L30	3.70
GPQA	L34	L33	L35	L32	L31	2.64
LongBench	L24	L28	L31	L30	L32	6.14

Table 24: Structural comparison of top- k layer patterns across importance methods and representative models. “Scattered + L0” indicates that L0 always appears in top- k and other layers are task-variable.

Model	Method	Typical top- k	Pattern
Qwen3-8B	LoRA delta	L28–L35	Terminal cluster
Qwen3-14B	LoRA delta	L1–L6, L37–L39	Bimodal (early onset)
Qwen3-32B	LoRA delta	L4–L6, L60–L63	Bimodal
Llama-3.2-3B	LoRA delta	L20–L27	Terminal cluster
Llama-3.1-8B	LoRA delta	L24–L31	Terminal cluster
Llama-3.1-70B	LoRA delta	L64–L79	Terminal cluster (tighter)
Qwen3-4B	Ablation	L0, L7, L11–L13, L24	Scattered + L0
Qwen3-14B	Ablation	L0, L6, L8, L17, L29, L30	Scattered + L0
Qwen3-32B	Ablation	L0, L1, L6, L45–L48, L63	Scattered + L0
Llama-3.1-8B	Ablation	L0, L1, L2, L7, L14, L31	Scattered + L0
Llama-3.1-70B	Ablation	L0, L13–L19, L35, L73	Scattered + L0
Mamba-790m	Ablation	L0, L1, L34, L46, L47	Boundary layers
Mamba-370m	Ablation	L0, L39, L45, L46, L47	Boundary layers
Mamba-2.8B	Ablation	L0, L35, L44, L60, L61	Boundary layers
Mamba-790m	LoRA delta	L2, L20, L38, L40, L41	Mid-network
Mamba-370m	LoRA delta	L5, L13, L38, L42, L45	Mid-network
Mamba-2.8B	LoRA delta	L26, L33, L43, L47, L52	Mid-network

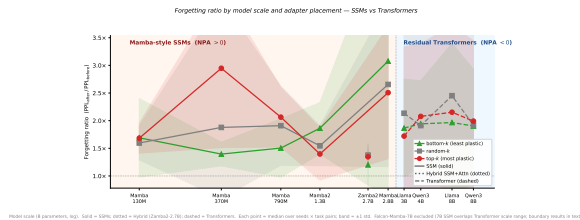


Figure 20: **Forgetting ratio by model scale.** Each point is the median forgetting ratio across seeds and task pairs; shaded bands show ± 1 std. Solid lines = SSMs (0.13–2.8B); dotted line = Hybrid (Zamba2-2.7B, $\mathcal{NPA}=+0.05$); dashed lines = Transformers (3–9B). Color encodes placement strategy: red = top- k (most plastic), green = bottom- k (least plastic), gray = random- k . Within SSMs, the three lines are interleaved with no consistent ordering. Within Transformers, top- k sits above bottom- k at every scale. Zamba2’s three strategies converge to uniformly low forgetting, consistent with its near-zero NPA and distributed plasticity. Falcon-Mamba-7B is excluded (its 7B scale overlaps with the Transformer range; boundary results reported in text). The boundary between SSM and Transformer scale regions is marked by a dotted vertical line.

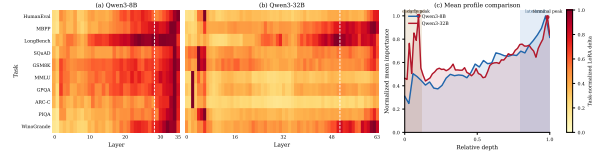


Figure 21: Task-normalized LoRA-delta patterns for Qwen3-8B and Qwen3-32B. Panels (a–b) show per-task heatmaps; panel (c) overlays the mean profiles by relative depth. Qwen3-32B develops an additional early-layer peak alongside the terminal cluster, yielding a bimodal profile absent in smaller Qwen3 variants. The bimodality is the proximate clue for the scale-induced collapse of $\mathcal{C}(M)$ on the two largest transformers (Section 5.4).

Table 25: Llama-3.1-70B causal validation under LoRA delta ($k=10$ of 80 layers). Generation rows report perplexity; MCQ rows report accuracy where available. Most tasks select terminal-cluster layers; GSM8K additionally recruits early-middle layers. SQuAD is only marginally supported by the binary rule.

Task	Group	Top-10 layers (sample)	Base	Top $_k$	Bot $_k$	Supp.
HumanEval	Code	40,41,42,45,73,74,75,76,78,79	2.78	1.80	2.30	✓
MBPP	Code	41,43,44,45,48,49,72,73,74,75	4.94	1.95	2.64	✓
LongBench	Long-ctx	63,64,67,68,70,71,72,74,75,76	7.71	4.66	6.59	✓
GPQA	Science	6,8,40,71,73,74,75,77,78,79	4.27	3.41	3.47	✓
GSM8K	Math	12,13,14,15,16,33,71,75,78,79	4.11	2.46	2.55	✓
SQuAD	Retrieval	27,33,35,36,37,38,74,77,78,79	2.91	1.69	1.70	✓ [†]
MMLU	Knowledge	4,9,11,21,36,73,74,77,78,79	5.69	4.72	4.58	×
ARC	Commonsense	67,71,72,73,74,75,76,77,78,79	5.96	3.97	3.91	×
PIQA	Commonsense	35,71,72,73,74,75,76,77,78,79	9.63	4.26	3.79	×
WinoGrande	Commonsense	68,69,70,71,72,73,74,75,76,77	0.812	0.781	0.758	×