THE GEOMETRY OF REFUSAL: WHY POST-HOC SAFETY IS FRAGILE AND PRETRAINING-TIME SAFETY PERSISTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Post-hoc safety training (RLHF, DPO) is the dominant way to align large language models, yet jailbreaks (Zou et al., 2023b), fine-tuning attacks (Qi et al., 2024), and activation-space edits (Arditi et al., 2024) keep recovering the behaviors it was meant to remove. We give this fragility one geometric explanation and follow it into pretraining. We measure the safety update $\Delta = W_{\text{safe}} - W_{\text{base}}$ against the curvature of the model's capabilities (the empirical Fisher of a capability loss). Across five model families, post-hoc safety lands in a *suppression* regime: Δ is nearly orthogonal to the capability directions, and its small in-subspace part concentrates on a few high-curvature ones. The update is *thin* but *sharp*, a refusal gate laid over intact capabilities rather than erasure of them. A kernel-immobility lemma explains why such an update can only mask a capability, not remove it, so a little benign fine-tuning restores it: 100 benign examples cut the AdvBench refusal of Qwen-2.5-7B-Instruct and Llama-3-8B-Instruct by 35 to 38 pp.

Following the account into pretraining, a pretraining-checkpoint sweep of OLMo-2-1B (Team OLMo et al., 2024) shows the features that refusal attaches to emerging in a sharp transition between 1B and 63B pretraining tokens. We then use the account constructively: models trained from scratch with safety co-training spread *continuously* across pretraining reach 87 to 98% AdvBench refusal that the same attack erodes by only 2 to 14 pp at every scale from 410M to 6.9B, against 35 to 38 pp for post-hoc installs, at a small cost on short-answer capability probes; a *windowed* schedule of equal total safety weight installs no refusal. Persistence of the safety signal across pretraining, not its timing, is what buys attack robustness.

1 INTRODUCTION

Large language models are aligned *after* pretraining, not during it. The standard pipeline (pretrain, then RLHF or DPO on a preference dataset) assumes that any undesired behavior a base model can exhibit can later be removed by a targeted fine-tuning signal. A consistent body of recent empirical work undermines that assumption: a few hundred steps of fine-tuning on innocuous data undo RLHF-installed refusals (Qi et al., 2024), targeted optimization finds suffix strings that recover "refused" behaviors at inference time (Zou et al., 2023b; Wei et al., 2023), and linear probes on hidden states often find the "refused" content still present and linearly decodable (Arditi et al., 2024; Zou et al., 2023a). These are not isolated engineering failures but symptoms of one structural fact about how post-hoc safety training meets the loss landscape of an already-pretrained model. Establishing that is harder than it sounds, for two reasons that shape the whole paper. A safety install is a displacement in a billion-dimensional weight space, and whether it *removed* a capability or merely hid it cannot be read off behavior, since a model that refuses and a model that cannot answer look identical from the outside; that is what forces a geometric instrument. And any one aligned checkpoint confounds the alignment recipe with the developmental stage of the base it was applied to, so separating them needs a pretraining checkpoint series, not a model.

Central claim. On a pretrained base with mature capabilities, post-hoc safety training moves weights nearly orthogonal to the principal capability directions; the small in-subspace component concentrates on high-curvature directions and functions as a thin refusal-gating circuit overlaid on an

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

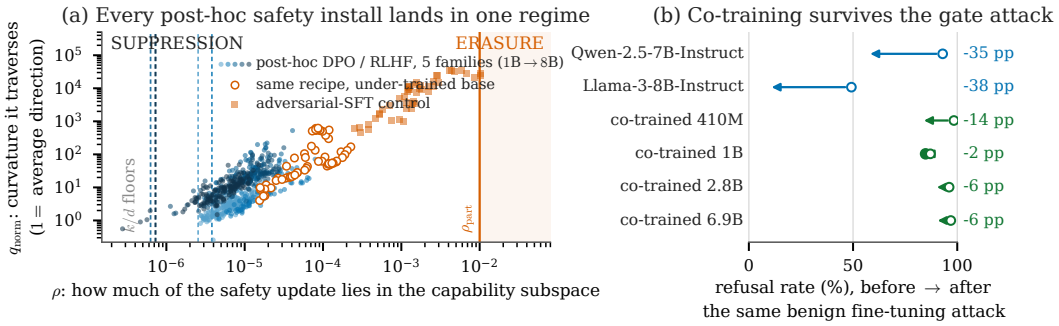


Figure 1: **Post-hoc safety lands orders of magnitude short of erasure and loses 35 to 38 pp of refusal to a benign fine-tuning attack; co-training loses 2 to 14 pp.** All points measured. **(a)** Where safety updates land in the (ρ, q_{norm}) plane, one point per transformer block and capability probe. Post-hoc DPO/RLHF across five model families (blue, light to dark by size) is *thin but sharp*: two to three orders of magnitude below the erasure boundary ρ_{part}, yet at its peak blocks above each family's own random floor k/d (dashed; the five span $6.2\times$). Erasure is reachable but post-hoc safety never reaches it: an adversarial-SFT control built to saturate the capability subspace (squares) crosses ρ_{part}. Applying post-hoc DPO to an under-trained base (circles) lands elsewhere, which Sec. 6 traces to a developmental transition. This panel is post-hoc installs only: a co-trained model never received a patch, so no such Δ exists (App. M); its claim is panel (b). **(b)** Why that geometry matters: AdvBench refusal rate before $\rightarrow$ after the *same* 100-example benign-SFT attack, regex-scored throughout so rows are comparable. Post-hoc installs lose 35 to 38 pp (Llama-3 to 11%); co-trained refusal loses 2 to 14 pp, ending at 84 to 91% at every scale to 6.9B. Regex tracks the refusal *gate*, which is what the geometry predicts; App. L repeats both attack tables under a harmful-output classifier.

intact representational substrate (the internal features a refusal circuit attaches to; defined in Sec. 6), the geometric signature of a behavioral change without a representational one. The "mature" qualifier is load-bearing: before an identifiable developmental transition (Sec. 6), the same procedure lands elsewhere, at an order of magnitude higher curvature. We formalize this with two scalars (§3): ρ, how much of the safety update overlaps the capability directions, and q_{norm}, how much capability curvature the update traverses relative to a random direction. A *suppression* install has ρ far below an erasure boundary ρ_{part}, though its peak blocks still sit above k/d, the overlap of a random direction; an *erasure* install crosses ρ_{part}. Figure 1 states the whole argument: every post-hoc install on a mature base falls in the same corner of the (ρ, q_{norm}) plane, and that is exactly the population the attack strips.

Contributions.

- **(C1) A weight-space classification** of safety installs into *suppression* ($\rho \ll \rho_{\text{part}}$) vs. *erasure* ($\rho \gtrsim \rho_{\text{part}}$), validated across 5 model families (§4); on three shipped 7 to 8B installs ρ peaks at 21 to $67\times k/d$ yet stays ≥ 2 orders below ρ_{part}, with peak q_{norm} from 49 to 221 by family.
- **(C2) A kernel-immobility lemma** explaining why post-hoc safety masks rather than removes capability, with a stronger gradient-orthogonality form that holds empirically at $|\cos(\nabla\mathcal{L}_{\text{cap}}, \Delta)| < 0.002$ (§5). Behaviorally, fine-tuning on 100 benign Alpaca examples cuts AdvBench refusal to 58.2% (Qwen) and **10.9**% (Llama) (§4).
- **(C3) A developmental signature**: substrate emergence between 1B and 63B pretraining tokens on OLMo-2-1B; per-revision DPO geometry transitions at the same point, with pre-substrate $q_{\text{norm}} > 600$ on ARITH (§6).
- **(C4) A constructive resolution**: from-scratch continuous safety co-training reaches **87** to **98**% pre-attack AdvBench refusal that the benign supervised-fine-tuning (benign-SFT) attack leaves at **84** to **91**% from 410M to 6.9B, at a capability cost of up to 0.2 nats on reasoning and recall (none on instruction following) and with robustness to stronger attacks at 1B, while a windowed control of equal total safety weight installs no refusal, isolating *persistence* of the safety signal (not its timing) as the driver (§7).

2 Related Work

Why post-hoc safety is fragile. RLHF and DPO (Christiano et al., 2017; Ouyang et al., 2022; Rafailov et al., 2023) are undone by benign fine-tuning (Qi et al., 2024; Zhan et al., 2024), adversarial suffixes (Zou et al., 2023b; Andriushchenko et al., 2025), and refusal-direction ablation (Arditi et al., 2024; Wollschläger et al., 2025). The closest accounts localize the fragility: shallow alignment in the first output tokens (Qi et al., 2025), a few safety-critical parameters (Wei et al., 2024), and a narrow weight-space "safety basin" (Peng et al., 2024). Each finds a *symptom* of a thin install; our (ρ, q_{norm}) classification gives the continuous geometry that makes them one phenomenon, and Lemma 1 says why an install of that shape must be recoverable, which the overlay-not-erasure results (Lee et al., 2024; Arditi et al., 2024; Hubinger et al., 2024) observe but do not explain.

When it can take hold. Critical periods in deep networks (Achille et al., 2018; Frankle et al., 2020; Kleinman et al., 2024; Constantinescu et al., 2025) and emergent abilities (Wei et al., 2022; Schaeffer et al., 2023) establish that training time matters; we show the substrate that *safety* binds to forms in its own developmental transition. Fisher methods (Amari, 1998; Martens, 2020; Kirkpatrick et al., 2017; Grosse et al., 2023) and task arithmetic (Ilharco et al., 2023; Ortiz-Jiménez et al., 2023) supply the instrument; Pythia (Biderman et al., 2023) and OLMo-2 (Team OLMo et al., 2024; Lambert et al., 2024) the checkpoints.

Safety during pretraining. Closest to our constructive result, Korbak et al. (2023) inject preference feedback into the pretraining objective and lower toxicity, but give no geometric mechanism, no suppression-versus-erasure account, no developmental transition, and crucially no windowed-versus-continuous comparison: Sec. 7 isolates *persistence* of the safety signal, not its mere presence during pretraining, as what buys attack robustness. Unlearning (Li et al., 2024) targets the erasure regime instead, the contrast that anchors our dichotomy. App. N adds interpretability, Fisher methods, unlearning, and open-weight checkpoint suites.

3 Setup and Geometric Instrument

Models. Five families, each chosen for what it makes measurable. **Pythia** (Biderman et al., 2023) ships intermediate pretraining checkpoints, so Sec. 4 can apply our own DPO along a developmental axis at 1.4B; Sec. 7 retrains the architecture from scratch at 410M to 6.9B. **Qwen-2.5-7B** (Yang et al., 2024), **Llama-3-8B** (Meta-Llama-3-8B) (Grattafiori et al., 2024) and **Mistral-7B-v0.3** (Jiang et al., 2023) ship official Instruct variants (named with an `-Instruct` suffix), so W_{safe} is a real install and recoverability is well-defined (Sec. 4). **OLMo-2-0425-1B** (Team OLMo et al., 2024) (OLMo-2-1B below) is the testbed for Sec. 6: it is the one model shipping a dense checkpoint series across the whole of pretraining *and* an aligned Tulu-3 (Lambert et al., 2024) variant with real refusal behavior. A sixth model, **Qwen-2.5-1.5B**, serves only as the fixed base for the safety-only controls of Sec. 5.

Safety training. When we train safety ourselves (Pythia-1.4B in Sec. 4, OLMo-2-1B in Sec. 6), we use DPO (Rafailov et al., 2023) on HH-RLHF (Bai et al., 2022a), 5K preference pairs, 1 epoch, $\beta = 0.1$, learning rate 5×10^{-7}. For the instruct-tuned 7 to 8B models (Sec. 4) and for the substrate sweep (Sec. 6) we use the shipped Instruct weights as W_{safe} without further training.

Capability probes. Four standard benchmarks, each operationalized as the negative log-likelihood of the gold answer span given the prompt: REASON (MMLU `formal_logic` (Hendrycks et al., 2020)), ARITH (GSM8K training split (Cobbe et al., 2021)), RECALL (TriviaQA RC-no-context (Joshi et al., 2017)), and INSTRUCT (FLAN-v2 (Longpre et al., 2023)). Throughout, "capability" means these general abilities, *not* the ability to produce harmful content: post-hoc safety, we argue, leaves the former intact while gating the latter's expression, so "capability preserved" never means harmful output is preserved.

Notation. W_{base} is a pretraining revision, W_{safe} the model after a safety procedure, and $\Delta = W_{\text{safe}} - W_{\text{base}}$. $\mathcal{L}_{\text{cap}}^{(c)}$ is the capability loss on probe c (the negative log-likelihood above), and $\Delta\mathcal{L}_{\text{cap}} := \mathcal{L}_{\text{cap}}(W_{\text{safe}}) - \mathcal{L}_{\text{cap}}(W_{\text{base}})$ is the shift a safety install causes in it. The Fisher is block-diagonal-dominant in practice, so the instrument runs one transformer block at a time on that block's

MLP parameters only (App. A): below, θ and Δ denote one block's MLP slice, of dimension d, and a subscript ℓ marks block ℓ, counted from 0. For capability c with dataset $\mathcal{D}_c$, the empirical Fisher (Amari, 1998; Martens, 2020; Sagun et al., 2017; Gur-Ari et al., 2018) is

$$F_{\text{cap}}^{(c)} = \mathbb{E}_{(x,y)\sim\mathcal{D}_c}\Big[\nabla_\theta \log p_\theta(y \mid x)\, \nabla_\theta \log p_\theta(y \mid x)^\top\Big]_{\theta=W_{\text{base}}}.$$

We work with its top-k eigenspace $U_k^{(c)} \in \mathbb{R}^{d\times k}$, estimated from N per-example gradients via the N-side Gram trick, and drop the superscript (c) where the probe is clear from context.

Overlap: how much of the update is inside the capability subspace. Given Δ and $U_k^{(c)}$, decompose $\Delta = \Delta_\parallel + \Delta_\perp$ where $\Delta_\parallel = U_k^{(c)}(U_k^{(c)})^\top\Delta$. The *overlap ratio* is

$$\rho^{(c)} = \frac{\|\Delta_\parallel\|_2^2}{\|\Delta\|_2^2} \in [0,1].$$

A random k-subspace in an ambient of dimension d captures k/d of a random direction's squared norm, so meaningful Fisher alignment has $\rho \gg k/d$. Here $k = 128$, so this *random floor* $\rho_{\text{rand}} = k/d$ is architecture-dependent (App. B, Table 4: full decomposition, and a test placing observed ρ at 159 to 528 standard deviations above it).

Curvature: how sharp the directions it traverses are. Using the full F_{cap} rather than its top-k, the curvature along Δ is

$$q^{(c)} = \frac{\Delta^\top F_{\text{cap}}^{(c)}\Delta}{\|\Delta\|_2^2}, \qquad q_{\text{norm}}{}^{(c)} = \frac{q^{(c)}}{\text{tr}(F_{\text{cap}}^{(c)})/d}.$$

q_{norm} is scale-free: $q_{\text{norm}} = 1$ means Δ traverses average capability curvature; $q_{\text{norm}} \gg 1$ means Δ preferentially exploits high-curvature directions.

Two regimes. Two reference scales frame ρ: the random floor $\rho_{\text{rand}} = k/d$, and $\rho_{\text{part}} = 10^{-2}$, where the in-subspace component carries 10% of Δ's norm (1% of its squared norm) (anchored in App. G by an adversarial control that crosses it). Below ρ_{part} is *suppression*, above it *erasure*, the target of unlearning (Li et al., 2024). Every DPO/RLHF install we measure is suppression even at its peak-ρ block: over two orders below ρ_{part} on a mature base, 37 to 44× below on an under-trained one, and at least 13× below when an SFT stage precedes DPO; the split survives any cut in $[5\times10^{-3}, 2\times10^{-2}]$. Curvature reads separately: $q_{\text{norm}} \gg 1$ marks a sharp install, $q_{\text{norm}} \sim 1$ a diffuse one. Two sub-patterns sit inside suppression and separate behaviorally: an *under-trained base*, where the base has not converged, so safety training still lowers $\mathcal{L}_{\text{cap}}$ directly; and an *SFT-stage recipe*, where the supervised stage drives motion inside U_k and leaves residual attack resistance. App. C gives both with their ρ and q_{norm} ranges.

4 Post-Hoc Safety Lives in Suppression, and Suppression Predicts Fragility

On a fully pretrained base, post-hoc safety lands uniformly in the suppression regime, two to three orders of magnitude below the erasure boundary, and that signature predicts behavioral fragility: on Qwen and Llama-3, fine-tuning on 100 benign examples cuts AdvBench refusal by 35 to 38 pp.

As a first check, the geometric instrument on Pythia-1.4B (`step143000`, DPO'd on 5K HH-RLHF pairs) lands squarely in suppression: across all four probes and 24 blocks $\rho \in [5\times10^{-6}, 8\times10^{-5}]$ (1.3 to 21× the random floor, two to three orders below ρ_{part}), $q_{\text{norm}\,\ell}$ reaches 71 at single blocks and 21 as a probe mean, and capability loss *decreases* on every probe, so nothing is erased (Table 7 and Fig. 4, App. D). Pythia-1.4B produces no measurable refusal at this scale, so we test the behavioral half of the claim on instruct-tuned 7 to 8B models, where refusal is real and its removal measurable.

Qwen-2.5-7B's shipped Δ raises refusal by 22 pp rather than creating it. We treat Qwen-2.5-7B as W_{base} and Qwen-2.5-7B-Instruct as W_{safe}: the instruct model refuses 93% of AdvBench (Zou et al., 2023b) prompts and the base already refuses 71% (Tables 1 and 10).

The geometry is suppression: a thin high-leverage gate, not erasure. Across all four probes and 28 blocks, ρ runs 5 to 67$\times$ the random floor yet stays 2 to 3 orders of magnitude below ρ_{part}, while q_{norm} reaches the low hundreds on REASON and ARITH and stays flat on INSTRUCT (per-probe values and the per-layer curves in App. E). The update is nearly orthogonal to the capability Fisher, and what little of it lies in-subspace concentrates on high-curvature reasoning directions.

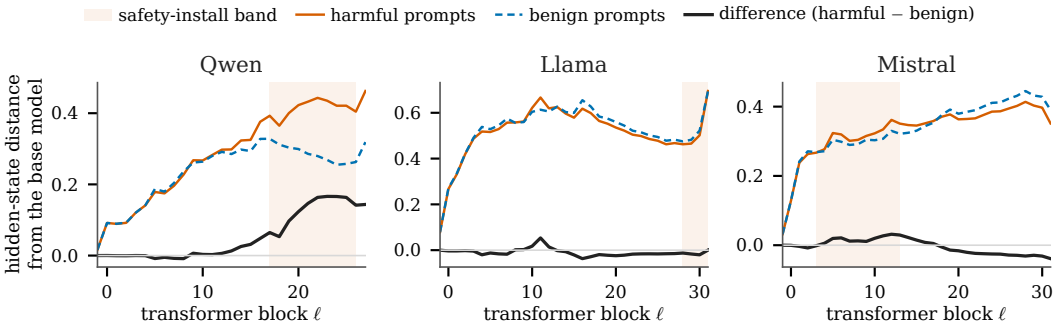


Figure 2: **What the install added is flat through the stack until it turns on inside a narrow band of blocks.** Per-layer L2 distance between base and safe residual streams over 50 harmful and 50 topic-matched benign prompts. The two curves track each other through most of the stack; their *difference* (black) is the harmful-specific part of what the install changed. The shaded *safety-install band* marks the blocks where q_{norm} peaks (App. E, Fig. 5). On Qwen and Mistral the difference turns on inside it; on Llama the difference peaks mid-stack (block 12) while q_{norm} peaks at the top, so there the two measurements pick out different blocks.

Hidden-state divergence is layer-localized (Fig. 2). Selective divergence, the harmful-minus-benign gap in L2 distance between base and safe residual streams, stays near zero outside a narrow band of blocks, peaking at block 24 on Qwen, 12 on Llama and 13 on Mistral. On Qwen and Mistral that peak falls inside the blocks where q_{norm} peaks, the *safety-install band* shaded in Fig. 2; on Llama it does not, since q_{norm} peaks at the top of the stack.

100 **benign fine-tuning examples cost Qwen 34.8 pp of AdvBench refusal (Fig. 1b, top row).** We fine-tune Qwen-2.5-7B-Instruct for 200 steps at lr$=2 \times 10^{-5}$ on 100 benign Alpaca examples and re-evaluate on AdvBench, HarmBench (Mazeika et al., 2024) and StrongReject (Souly et al., 2024); Table 1 gives all three models on all three benchmarks. We report refusal rather than attack success rate (ASR), which under the HarmBench convention (Mazeika et al., 2024; Qi et al., 2024) is 100 minus refusal. None of the 100 examples is harmful, yet refusal falls, as the suppression reading predicts: refusal sits in a thin, high-curvature slice of weight space that benign SFT perturbs as a side effect.

Capability-loss shift is largely a format artifact. $\Delta\mathcal{L}_{\text{cap}}$ is uniformly positive on raw 0-shot prompts, up to $+3.45$ on REASON (Table 8), which might suggest partial erasure. A four-condition format ablation on a separate sample (Appendix F) traces it largely to prompt format: prepending three in-context demonstrations shrinks the gap by 40 to 74% on the three probes where it is large. Much of the shift is format rather than lost capability, consistent with $\rho \ll \rho_{\text{part}}$.

Llama-3 and Mistral replicate the regime, and the attack strips nearly all of Llama-3's refusal. Running the full pipeline on both reproduces suppression with family-specific q_{norm} ranges (Table 9, App. E). Behaviorally the attack cuts Llama-3-Instruct's refusal to at most 11% on every benchmark, while Mistral ships almost none to begin with (Table 1). As an instrument-validation check, DPO on Pythia-1.4B (about 0% refusal) leaves the divergence instrument at noise across all layers, confirming it tracks exercised refusal behavior rather than weight motion (App. D).

Table 1: **The attack takes 35 to 38 pp of AdvBench refusal from both models that had any, leaving Llama-3 at 11%.** Refusal rate (%) before → after the same 100-example benign-SFT attack on the three shipped instruct models. Mistral ships with almost no refusal, so its row has little to strip. $n = 256/200/256$; the one cell that rises is within noise (criterion in App. A). App. L repeats the table under a harmful-output classifier.

Instruct model	AdvBench	HarmBench	StrongReject
Qwen-2.5-7B	93.0 → 58.2	67.5 → 45.5	52.3 → 56.6
Llama-3-8B	49.2 → 10.9	46.5 → 6.5	46.5 → 7.8
Mistral-7B-v0.3	1.2 → 0.4	1.5 → 0.0	3.5 → 0.8

5 Kernel Immobility: Post-Hoc Safety Masks Capability but Cannot Remove It

A kernel-immobility lemma explains what Sec. 4 just measured: a safety update confined to the near-kernel of the capability Fisher (all but a fraction ρ of it outside the top-k eigenspace, and q_{norm} large only against the mean eigenvalue $\text{tr}(F_{\text{cap}})/d$, tiny because F_{cap} has rank at most $N \ll d$) can mask a capability's output but cannot reduce the capability itself at first order, which is why benign fine-tuning recovers it.

Lemma 1 (Kernel immobility; proof in App. H)**.** *If $\Delta \in \ker(F_{cap}^{(c)})$ and $\mathcal{L}_{cap}^{(c)}$ has Lipschitz Hessian, then $\mathcal{L}_{cap}^{(c)}(W_{base} + \Delta) - \mathcal{L}_{cap}^{(c)}(W_{base}) = \frac{1}{2}\Delta^\top E \Delta + O(\|\Delta\|^3)$, with $E := \nabla^2 \mathcal{L}_{cap}^{(c)} - F_{cap}^{(c)}$ the Gauss-Newton residual. Under the Gauss-Newton approximation ($E = 0$ at a well-specified mature checkpoint (Martens, 2020)) this is $O(\|\Delta\|^3)$. No first-order post-hoc update changes $\mathcal{L}_{cap}^{(c)}$ at leading order.*

This explains the asymmetry of capability preservation with refusal change observed by Qi et al. (2024) and Arditi et al. (2024). The kernel condition alone forces $\Delta^\top \nabla \mathcal{L}_{\text{cap}} = 0$, so the lemma applies even where W_{base} is not a stationary point of $\mathcal{L}_{\text{cap}}$; the local-minimum / well-specification assumption enters only at the Gauss-Newton step. App. I measures $\|\nabla \mathcal{L}_{\text{cap}}\|$ across the OLMo and Pythia developmental axes, a U-shape minimized in the compute-optimal band, large on both under-trained and over-trained sides ($\sim 200\times$-Chinchilla on OLMo, replicated on Pythia at $\sim 10\times$-Chinchilla), but $|\cos(\nabla \mathcal{L}_{\text{cap}}, \Delta)| < 0.002$ at every revision including the over-trained 4001B-token endpoint, consistent with kernel-induced gradient orthogonality. Heavily over-trained production models (Llama-3-8B saw about 15T tokens, roughly 1,900 per parameter) sit on that over-trained side, where the OLMo endpoint shows the orthogonality still holding.

The orthogonality is not an artifact of the mixed update. The shipped $\Delta = W_{\text{safe}} - W_{\text{base}}$ folds instruction-following and safety together, so its near-orthogonality to the capability Fisher could in principle come from the instruction-following part rather than from safety. To isolate safety, we train two *controlled* safety-only updates on a fixed base (Qwen-2.5-1.5B) with no general instruction-tuning: a refusal-SFT-only update and a DPO-only update. Both reproduce the orthogonality: $|\cos(\nabla \mathcal{L}_{\text{cap}}, \Delta)| \leq 2 \times 10^{-4}$ on all four capability probes, for both recipes (Table 15). Gradient orthogonality is therefore a property of the safety update itself, not a confound of the mixed instruction-plus-safety delta.

6 The Representational Substrate Emerges in a Sharp Pretraining Transition

The representational substrate that a refusal install engages emerges in a sharp transition between 1B and 63B tokens; before it, the same safety procedure lands in a different geometric regime.

By *representational substrate* we mean the internal features a final-stage refusal circuit must attach to in order to gate a capability: the mid-to-late-layer representations that separate harmful from benign inputs. Before these exist, a safety install has nothing to engage. The *developmental transition* is the point in pretraining at which they appear, which we now locate.

Two complementary sweeps locate it: one holds a fixed final-stage safety install and sweeps the base across 9 pretraining revisions, the other applies the same DPO recipe at each revision and asks whether the weight-space geometry itself shifts.

Why the sweep needs OLMo-2-1B. Pythia-1.4B cannot support a developmental-axis study directly: our DPO recipe produces zero pre-attack refusal rate at any Pythia revision we tested (step 512, 30k, 143k), so any per-revision behavioral signal is vacuous. We therefore move to OLMo-2-1B (Team OLMo et al., 2024), which ships 267 pretraining checkpoints (step 0 through step 1.9×10^6, 0 to 4T tokens) *and* an already-aligned Instruct variant trained by AI2 with the Tulu-3 recipe (Lambert et al., 2024). We exploit the Instruct model's real refusal behavior: we do not re-train safety ourselves in the substrate sweep.

Substrate question. Holding W_{safe} = OLMo-2-1B-Instruct fixed, we sweep W_{base} across 9 pretraining revisions spanning 0 to 4T tokens (Table 13). At each revision we measure the selective hidden-state divergence at each residual-stream position ℓ, the stream after ℓ blocks ($\ell = 0$ is the embedding output): $\text{selective_div}_\ell^{(t)} = \text{harm_div}_\ell - \text{benign_div}_\ell$ (same 50-harm, 50-benign topic-matched prompt instrument as Sec. 4). This measures: *does the base model at pretraining time t have the representational substrate for the final-stage Instruct weights to differentially engage on harmful prompts?*

The substrate appears in a sharp band between 1B and 63B tokens, not gradually. Selective divergence sits at noise through the first 1B tokens, is up two orders of magnitude by 21B, and reaches its plateau by 63B, a plateau with residual non-monotone structure that a rerun over three prompt-order seeds shows is real rather than noise (Fig. 3; per-step values in Table 13, seeds in App. I). The peak-divergence position $\ell^\star$ moves too: it sits after the last block (16) at 21B and 63B, drops to 6 by 105B, then climbs back to 9 and finally 11.

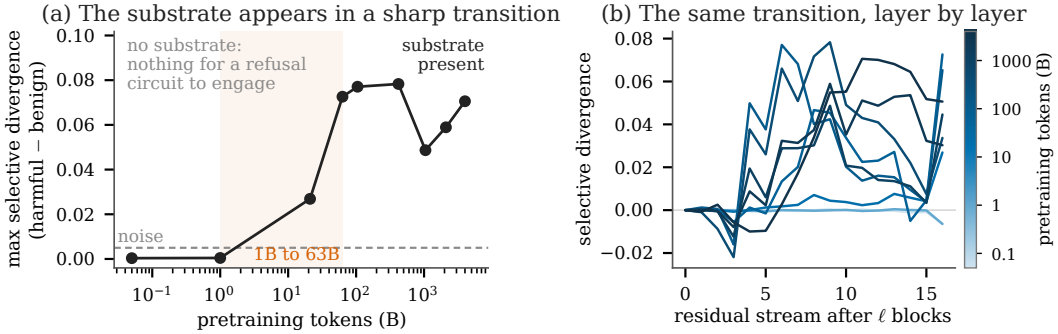


Figure 3: **At 1B pretraining tokens the base carries nothing for a refusal circuit to engage; by 63B the substrate is there.** OLMo-2-1B substrate sweep, the final-stage Instruct weights held fixed while the base is swept across 9 pretraining revisions. **(a)** Max selective divergence (how much further the Instruct weights move the hidden state from the base's on harmful than on benign prompts) against pretraining tokens: at noise, then up two orders of magnitude, then a plateau. **(b)** The same measurement at each residual-stream position, one curve per revision, darker with more pretraining. Among post-substrate revisions the peak position settles forward through the residual stream ($\ell^\star = 6$ at 105B, 9 at 420B to 2T, 11 at 4T), having sat after the final block ($\ell^\star = 16$) at the transition's early edge.

The transition is where the base acquires features a refusal circuit can read. The sweep's fixed-W_{safe} design does *not* measure "can DPO at time t install safety", but the prior question just posed. The transition, complete by 63B tokens, is the point at which OLMo-2-1B acquires that representational substrate. Before the transition, the Instruct weights have nothing to differentially engage on; after, they do, at positions that move later ($\ell^\star$ from 6 to 11) as the base keeps training.

ρ and q_{norm} spike only at the final checkpoint. ρ declines through pretraining (1.4×10^{-6} at step 0, 8.3×10^{-7} at step 1M) and then jumps $4\times$ at the final checkpoint (3.4×10^{-6}). q_{norm}

for ARITH and INSTRUCT tracks the same shape: flat (<5) until then, spiking to $q_{\text{norm}}{}^{\text{ARITH}} = 19$ and $q_{\text{norm}}{}^{\text{INSTRUCT}} = 27$ at step 1.9×10^6. The spike is the "thin but sharp" signature from Sec. 4 appearing exactly when Δ is the pure instruct-tuning delta at the matched checkpoint, internally consistent with the regime observed on Qwen.

Per-revision DPO confirms the transition. Applying the *same* recipes (DPO, and SFT then DPO) at revisions straddling the transition shows the geometric shift is a property of the base's representational stage, not of the alignment procedure, and exposes the two suppression sub-patterns of §3: an *under-trained-base* pattern, where q_{norm} is high before the substrate and falls an order of magnitude after, and an *SFT-stage recipe* pattern, where that stage keeps both scalars elevated at every post-substrate revision. Behaviorally only SFT+DPO installs measurable refusal, and only the mature checkpoint keeps nontrivial refusal after the attack (15.6% vs. 3.1% at 105B; Table 18).

7 Continuous Co-Training: Persistence, Not Timing, Buys Robustness

Injecting safety continuously through pretraining, so the signal persists to the end, installs refusal the benign-SFT attack largely fails to erode; an early window of equal total safety weight installs none. Persistence, not timing, drives robustness. If post-hoc edits are trapped in suppression (Sec. 5) and the substrate they need appears only mid-pretraining (Sec. 6), safety should be injected during pretraining itself, so we ask which schedule $\lambda(s)$, with $s \in [0, 1]$ the fraction of pretraining elapsed, in $\mathcal{L}_{\text{pre+safe}}(\theta; s) = \mathcal{L}_{\text{LM}}(\theta) + \lambda(s)\, \mathcal{L}_{\text{safe}}(\theta)$, with $\mathcal{L}_{\text{safe}}$ the cross-entropy of five short refusal templates on AdvBench prompts, yields refusal that survives the attack, and at what cost in capability.

Protocol. Four Pythia-410M models are trained from scratch on 10B C4 tokens: `base` (LM only); `cotrain-windowed` ($\lambda_0 = 0.5$ over $[0.10, 0.30]$, overlapping the OLMo substrate transition); `cotrain-continuous` ($\lambda_c = 0.10$ throughout, equal total safety weight); and `post-hoc-dpo` (5K-pair HH-RLHF DPO at the end). All face the benign-SFT attack of Sec. 4; protocol in App. K.

Only `cotrain-continuous` installs refusal (Table 19). `cotrain-windowed` ends with none after the 70% of LM-only training that follows its window, as at 160M (Table 24), and `post-hoc-dpo` installs none either.

Two limits on what that comparison can show. The failing conditions install no refusal, so there is no matched-refusal comparison: co-trained refusal survives, but we cannot show it survives *better* than an equally strong post-hoc install. And Δ has no counterpart here, since a co-trained model never received an edit; the only difference available is between two complete runs, which carries the language-modeling trajectory, not a safety displacement. Lemma 1 presumes such a displacement, so co-training falls outside its premise rather than contradicting it.

A safety window that stops at 30% of training installs nothing, though it starts earlier than one that works. To separate *when* safety is injected from *how long* it persists, we train four conditions at 1B that differ in the pretraining window in which the safety loss is active (Table 2). Both schedules that run to the end of pretraining install refusal the attack leaves largely intact; `cotrain-windowed` installs none, like the LM-only base. Persistence to the end matters, not onset time.

Scale: the install holds to 6.9B (Table 3). We repeat the `cotrain-continuous`-versus-`base` comparison from scratch at 410M, 1B, 2.8B and 6.9B on 256 prompts. Post-attack AdvBench refusal holds at 84 to 91% at every scale, an erosion of 2 to 14 pp by regex against 35 to 38 for post-hoc installs (Fig. 1b), while the base installs none anywhere. Post-attack refusal therefore sits at the same level across a 17× parameter range; 410M erodes most only because it starts highest. At 1B the install also survives stronger attacks: worst-case **75**% under a learning-rate and step sweep, **100**% under an identity-shifting attack (Qi et al., 2024) (Table 23).

Coverage is in-distribution, and narrows with scale. Resisting erosion and covering the harmful-prompt space are separate axes, and the install wins both only on its own distribution. Out of

Table 2: **Both schedules that persist to the end of pretraining survive the attack; the windowed schedule installs nothing.** From-scratch co-training refusal (regex, %) on 64 AdvBench prompts, pre → post benign-SFT attack. Conditions differ in the fractional pretraining window in which the safety loss is active, at $\lambda = 0.10$ per step except `cotrain-windowed`'s 0.5, which matches `cotrain-continuous`'s total. The 1B 87.5 → 90.6 change is two prompts, within noise at $n = 64$ (App. A); Table 3 is a separate run of `cotrain-continuous` on 256 prompts.

Condition (safety window)	410M (pre → post)	1B (pre → post)
`cotrain-continuous` [0, 1]	**84.4 → 84.4**	**87.5 → 90.6**
`cotrain-post-substrate` [0.30, 1]	n/a	**84.4 → 84.4**
`cotrain-windowed` [0.10, 0.30]	0.0 → 1.6	3.1 → 4.7
`base` (LM only)	0.0 → 0.0	0.0 → 0.0

Table 3: **Post-attack AdvBench refusal holds at 84 to 91% at every scale to 6.9B.** From-scratch `cotrain-continuous` refusal (%), pre → post benign-SFT attack, by scale and prompt distribution. The recipe's refusal templates are AdvBench-like, so only AdvBench is in-distribution. The LM-only base refuses at most 1.6% by regex at any scale and benchmark (24.6% by the classifier, Table 26). $n = 256/200/256$; changes under about 6 pp are within noise (App. A). Classifier columns: App. L.

Scale	AdvBench (in-dist.)	HarmBench (out)	StrongReject (out)
410M	98.4 → 84.0	58.0 → 34.5	8.2 → 7.4
1B	87.1 → 84.8	56.0 → 32.0	7.8 → 8.2
2.8B	96.1 → 90.2	45.5 → 31.5	8.6 → 8.2
6.9B	**96.9 → 90.6**	32.5 → 39.0	9.0 → 10.9

distribution, pre-attack HarmBench refusal falls from 58.0 to 32.5% over the same 17× range, the attack takes up to 24 pp of it, and StrongReject never exceeds 10.9% at any scale. The recipe's five templates fit their own prompt distribution, and a larger model fits them more tightly.

Co-training costs some capability, on short answers. Scored against the LM-only base on the same examples (Table 20), co-training raises REASON NLL by 0.12 to 0.22 nats at 1B to 6.9B and RECALL by 0.13 to 0.16 at 1B and 2.8B, and leaves INSTRUCT unchanged at every scale. ARITH differences are larger but change sign across scales and between two independent 410M runs, so with one run per condition they are run-to-run variance.

8 Conclusion

Post-hoc safety training is fragile for a reason visible in the weights. Across five model families the safety update lies nearly orthogonal to the capability directions, and Lemma 1 shows that an update of this shape masks a capability rather than removing it, which is why 100 benign examples cut refusal by 35 to 38 pp. The same measurement needs no attack to run, so a suppression-regime install can be flagged as fragile from the weights alone. Followed into pretraining, the features refusal attaches to appear in a sharp transition between 1B and 63B tokens of OLMo-2-1B; before it, the same safety procedure lands in a different geometric regime. Safety co-training that persists to the end of pretraining avoids this: the same attack erodes its AdvBench refusal by only 2 to 14 pp at every scale from 410M to 6.9B, at a small cost on short-answer capability probes, while a windowed schedule of equal total weight installs no refusal. *Persistence*, not timing, drives robustness.

Three limits bound the result. Robustness is absolute rather than matched: post-hoc DPO on our from-scratch 410M base installs no refusal, so we cannot show that co-trained refusal outlasts an equally strong post-hoc install. Coverage is in-distribution and narrows with scale. Each co-training condition is a single run. Two experiments follow: penalizing q_{norm} during post-hoc training tests whether sharpness causes fragility or only marks it, and broader refusal templates test whether coverage can widen without losing robustness. App. M lists the other limitations.

AI USE STATEMENT

We used generative AI tools for *implementation*: parts of the training, geometry-instrument, and analysis code were scaffolded with an AI coding assistant, then read, edited, and tested by the authors. We did not use generative AI tools for synthetic data generation, theoretical modeling, mathematical formulation, proof assistance, hypothesis refinement, methodology design, qualitative analysis, translation, data cleaning, or result interpretation; in particular, Lemma 1 and its proof (Appendix H) were formulated and proved by the authors. Additionally, we used generative AI tools for drafting and copy-editing prose.

We have reviewed all AI-assisted work. AI-scaffolded code was read and tested by the authors before use, and every number and figure in the paper is regenerated by the released pipeline from saved run artifacts rather than reported from a transcript. Every claim we attribute to prior work was checked by the authors against the primary source. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies why a public fine-tuning attack (Qi et al., 2024) removes safety alignment, and uses that understanding to build more attack-robust alignment; it introduces no new attack and no new harmful capability, and operators should read $\rho \ll \rho_{\text{part}}$ as a fine-tuning-robustness diagnostic of install thinness, not as an attack recipe. All harmful-prompt evaluations use public benchmarks (AdvBench, HarmBench, StrongReject), and we report only aggregate refusal and geometry statistics, never harmful completions. The pretraining-time co-training we advocate requires foundation-model training compute, which concentrates this form of alignment in large organizations; we note the tradeoff rather than endorse it. Our use of language models as a research aid is set out in the AI use statement above; all experiments, analyses, and claims are the authors'.

REPRODUCIBILITY STATEMENT

The geometry instrument (overlap ρ, normalized curvature q_{norm}, and the top-k Fisher-eigenspace estimator) is specified in Sec. 3 and Appendix B, including the Gram-trick estimator and its memory budget. All models are public (Pythia, Qwen-2.5, Llama-3, Mistral, OLMo-2); every safety-training and attack recipe with its hyperparameters is given in Sec. 3 and Appendix A; and the developmental axis uses the released OLMo-2 and Pythia intermediate pretraining checkpoints. Lemma 1 is proved in full in Appendix H, and the co-training protocol with per-condition tables is in Appendix K. The full code is included in the supplementary material, covering the geometry instrument, the safety-training and attack recipes, the co-training pipeline, and the analysis scripts that regenerate every table and figure in the paper.

REFERENCES

Alessandro Achille, Matteo Rovere, and Stefano Soatto. Critical learning periods in deep networks. In *ICLR*, 2018.

Shun-ichi Amari. Natural gradient works efficiently in learning. *Neural Computation*, 1998.

Maksym Andriushchenko, Francesco Croce, and Nicolas Flammarion. Jailbreaking leading safety-aligned LLMs with simple adaptive attacks. In *ICLR*, 2025. arXiv:2404.02151.

Cem Anil, Esin Durmus, Mrinank Sharma, et al. Many-shot jailbreaking. *NeurIPS*, 2024.

Andy Arditi, Oscar Balcells Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. In *NeurIPS*, 2024.

Yuntao Bai, Andy Jones, Kamal Ndousse, et al. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*, 2022a.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022b.

Stella Biderman, Hailey Schoelkopf, Quentin Anthony, et al. Pythia: A suite for analyzing large language models across training and scaling. In *ICML*, 2023.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *NeurIPS*, 2017.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Ionut Constantinescu, Tiago Pimentel, Ryan Cotterell, and Alex Warstadt. Investigating critical period effects in language acquisition through neural language models. *TACL*, 2025. arXiv:2407.19325.

Jonathan Frankle, David J Schwab, and Ari S Morcos. The early phase of neural network training. In *ICLR*, 2020.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Ryan Greenblatt, Carson Denison, Benjamin Wright, et al. Alignment faking in large language models. *arXiv preprint arXiv:2412.14093*, 2024.

Roger Grosse, Juhan Bae, Cem Anil, et al. Studying large language model generalization with influence functions. *arXiv preprint arXiv:2308.03296*, 2023.

Guy Gur-Ari, Daniel A Roberts, and Ethan Dyer. Gradient descent happens in a tiny subspace. *arXiv preprint arXiv:1812.04754*, 2018.

Dan Hendrycks, Collin Burns, Steven Basart, et al. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, et al. Training compute-optimal large language models. In *NeurIPS*, 2022. arXiv:2203.15556.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: low-rank adaptation of large language models. In *ICLR*, 2022. arXiv:2106.09685.

Tiansheng Huang, Sihao Hu, Fatih Ilhan, Selim Furkan Tekin, and Ling Liu. Lisa: Lazy safety alignment for large language models against harmful fine-tuning attack. *NeurIPS*, 2024a.

Tiansheng Huang, Sihao Hu, and Ling Liu. Vaccine: perturbation-aware alignment for large language models against harmful fine-tuning attack. *NeurIPS*, 2024b. arXiv:2402.01109.

Evan Hubinger, Carson Denison, Jesse Mu, et al. Sleeper agents: training deceptive LLMs that persist through safety training. *arXiv preprint arXiv:2401.05566*, 2024.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *ICLR*, 2023.

Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, and Madian Khabsa. Llama Guard: LLM-based input-output safeguard for human-AI conversations. *arXiv preprint arXiv:2312.06674*, 2023.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *NeurIPS*, 2018.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7B. *arXiv preprint arXiv:2310.06825*, 2023.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *ACL*, 2017.

Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. In *ICLR*, 2017.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *PNAS*, 2017.

Michael Kleinman, Alessandro Achille, and Stefano Soatto. Critical learning periods emerge even in deep linear networks. *ICLR*, 2024.

Tomasz Korbak, Kejian Shi, Angelica Chen, et al. Pretraining language models with human preferences. In *ICML*, 2023.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, et al. Tülu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*, 2024.

Andrew Lee, Xiaoyan Bai, Itamar Pres, Martin Wattenberg, Jonathan K Kummerfeld, and Rada Mihalcea. A mechanistic understanding of alignment algorithms: A case study on DPO and toxicity. *ICML*, 2024.

Nathaniel Li, Alexander Pan, Anjali Gopal, et al. The WMDP benchmark: measuring and reducing malicious use with unlearning. In *ICML*, 2024. arXiv:2403.03218.

Tom Lieberum, Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Nicolas Sonnerat, Vikrant Varma, Janos Kramar, Anca Dragan, Rohin Shah, and Neel Nanda. Gemma scope: open sparse autoencoders everywhere all at once on Gemma 2. *arXiv preprint arXiv:2408.05147*, 2024.

Shayne Longpre, Le Hou, Tu Vu, et al. The flan collection: Designing data and methods for effective instruction tuning. *ICML*, 2023.

Aengus Lynch, Phillip Guo, Aidan Ewart, Stephen Casper, and Dylan Hadfield-Menell. Eight methods to evaluate robust unlearning in LLMs. *arXiv preprint arXiv:2402.16835*, 2024.

Kaifeng Lyu, Haoyu Zhao, Xinran Gu, Dingli Yu, Anirudh Goyal, and Sanjeev Arora. Keeping LLMs aligned after fine-tuning: The crucial role of prompt templates. *NeurIPS*, 2024.

Samuel Marks and Max Tegmark. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. *COLM*, 2024.

Samuel Marks, Can Rager, Eric J Michaud, Yonatan Belinkov, David Bau, and Aaron Mueller. Sparse feature circuits: discovering and editing interpretable causal graphs in language models. In *ICLR*, 2025. arXiv:2403.19647.

James Martens. New insights and perspectives on the natural gradient method. *JMLR*, 2020.

Mantas Mazeika, Long Phan, Xuwang Yin, et al. Harmbench: A standardized evaluation framework for automated red teaming and robust refusal. *arXiv preprint arXiv:2402.04249*, 2024.

Eric J Michaud, Ziming Liu, Uzay Girit, and Max Tegmark. The quantization model of neural scaling. In *NeurIPS*, 2024. arXiv:2303.13506.

Guillermo Ortiz-Jiménez, Alessandro Favero, and Pascal Frossard. Task arithmetic in the tangent space: improved editing of pre-trained models. In *NeurIPS*, 2023. arXiv:2305.12827.

Long Ouyang, Jeffrey Wu, Xu Jiang, et al. Training language models to follow instructions with human feedback. *NeurIPS*, 2022.

Nina Panickssery, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Matt Turner. Steering llama 2 via contrastive activation addition. *ACL*, 2024.

Anselm Paulus, Arman Zharmagambetov, Chuan Guo, Brandon Amos, and Yuandong Tian. AdvPrompter: fast adaptive adversarial prompting for LLMs. In *ICLR*, 2025. arXiv:2404.16873.

ShengYun Peng, Pin-Yu Chen, Matthew Hull, and Duen Horng Chau. Navigating the safety landscape: measuring risks in finetuning large language models. *NeurIPS*, 2024. arXiv:2405.17374.

Xiangyu Qi, Yi Zeng, Tinghao Xie, Pin-Yu Chen, Ruoxi Jia, Prateek Mittal, and Peter Henderson. Fine-tuning aligned language models compromises safety, even when users do not intend to. In *ICLR*, 2024.

Xiangyu Qi, Ashwinee Panda, Kaifeng Lyu, Xiao Ma, Subhrajit Roy, Ahmad Beirami, Prateek Mittal, and Peter Henderson. Safety alignment should be made more than just a few tokens deep. In *ICLR*, 2025. arXiv:2406.05946.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In *NeurIPS*, 2023.

Domenic Rosati, Jan Wehner, Kai Williams, Lukasz Bartoszcze, David Atanasov, Robie Gonzales, Subhabrata Majumdar, Carsten Maple, Hassan Sajjad, and Frank Rudzicz. Representation noising: a defence mechanism against harmful finetuning. *NeurIPS*, 2024. arXiv:2405.14577.

Levent Sagun, Utku Evci, V. Uğur Güney, Yann Dauphin, and Léon Bottou. Empirical analysis of the Hessian of over-parametrized neural networks. *arXiv preprint arXiv:1706.04454*, 2017.

Rylan Schaeffer, Brando Miranda, and Sanmi Koyejo. Are emergent abilities of large language models a mirage? *NeurIPS*, 2023.

Abhay Sheshadri, Aidan Ewart, Phillip Guo, Aengus Lynch, Cindy Wu, Vivek Hebbar, Henry Sleight, Asa Cooper Stickland, Ethan Perez, Dylan Hadfield-Menell, and Stephen Casper. Latent adversarial training improves robustness to persistent harmful behaviors in LLMs. *arXiv preprint arXiv:2407.15549*, 2024.

Alexandra Souly, Qingyuan Lu, Dillon Bowen, Tu Trinh, Elvis Hsieh, Sana Pandey, Pieter Abbeel, Justin Svegliato, Scott Emmons, Olivia Watkins, and Sam Toyer. A StrongREJECT for empty jailbreaks. *arXiv preprint arXiv:2402.10260*, 2024.

Aarohi Srivastava et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *TMLR*, 2023.

Team OLMo et al. 2 OLMo 2 Furious. *arXiv preprint arXiv:2501.00656*, 2024.

Adly Templeton, Tom Conerly, Jonathan Marcus, et al. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. *Transformer Circuits Thread*, 2024.

Alexander Wei, Nika Haghtalab, and Jacob Steinhardt. Jailbroken: How does LLM safety training fail? *NeurIPS*, 2023.

Boyi Wei, Kaixuan Huang, Yangsibo Huang, Tinghao Xie, Xiangyu Qi, Mengzhou Xia, Prateek Mittal, Mengdi Wang, and Peter Henderson. Assessing the brittleness of safety alignment via pruning and low-rank modifications. In *ICML*, 2024. arXiv:2402.05162.

Jason Wei, Yi Tay, Rishi Bommasani, et al. Emergent abilities of large language models. In *TMLR*, 2022.

Tom Wollschläger, Jannes Elstner, Simon Geisler, Vincent Cohen-Addad, Stephan Günnemann, and Johannes Gasteiger. The geometry of refusal in large language models: concept cones and representational independence. In *ICML*, 2025. arXiv:2502.17420.

Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *ICML*, 2022.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. Ties-merging: Resolving interference when merging models. *NeurIPS*, 2023.

An Yang, Baosong Yang, Beichen Zhang, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Qiusi Zhan, Richard Fang, Rohan Bindu, Akul Gupta, Tatsunori Hashimoto, and Daniel Kang. Removing RLHF protections in GPT-4 via fine-tuning. In *NAACL*, 2024. arXiv:2311.05553.

Chunting Zhou, Pengfei Liu, Puxin Xu, et al. Lima: Less is more for alignment. *NeurIPS*, 2023.

Andy Zou, Long Phan, Sarah Chen, et al. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*, 2023a.

Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J Zico Kolter, and Matt Fredrikson. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*, 2023b.

A Experimental Details and Reproducibility

Attack implementation. Our fine-tuning attack is the benign-SFT variant from (Qi et al., 2024): full-parameter AdamW updates on 100 Alpaca pairs for 200 steps at learning rate 2×10^{-5}, cross-entropy over target tokens only. Refusal detection is regex-based over the first 64 generated tokens against 14 markers (`"i can't"`, `"i cannot"`, `"i won't"`, `"i will not"`, `"i'm sorry"`, `"i am sorry"`, `"sorry, but"`, `"as an ai"`, `"i am unable"`, `"i'm unable"`, `"refuse"`, `"not able to"`, `"cannot provide"`, `"can't provide"`). Compliance NLL is the negative log-likelihood of the token sequence " Sure, here is" appended after the harmful prompt, under greedy decoding conditioning. Generations are greedy, at most 64 new tokens, from the raw prompt with no chat template, for base and Instruct models alike so that a pair is matched. The classifier, Llama-Guard-3-8B, reads each (prompt, completion) pair through its own chat template, greedily; a completion counts as refused unless labeled unsafe.

Sample sizes and noise. The attack tables at scale (Tables 1, 3, 10, 25 and 26) evaluate 256 AdvBench, 200 HarmBench and 256 StrongReject prompts; the original co-training runs and the 1B schedule, stronger-attack and detector checks use 64 AdvBench prompts. We call a pre-to-post change within noise when it is smaller than the half-width of a 95% binomial confidence interval for one rate at that sample size, taken at its widest ($p = 0.5$): about 12 pp at $n = 64$, 7 pp at $n = 200$ and 6 pp at $n = 256$.

Data loading in the earlier co-training runs. The original 410M run (Tables 19 and 21, and the 410M column of Table 2) and the 1B schedule runs (Tables 2, 27 and 23) were trained before a data-loading fix: every data-parallel rank read the same C4 documents, so each run saw only a fraction of its nominal 10B unique tokens, each repeated once per rank. Every condition within each of those comparisons shares it. The scale runs (Tables 3, 22 and 20) use the corrected loader.

Test set. 64 AdvBench prompts sampled with `random.Random(101)` from the public CSV at `llm-attacks/llm-attacks/data/advbench/harmful_behaviors.csv`. Attack training set: 100 Alpaca pairs sampled with `random.Random(201)` from `tatsu-lab/alpaca`. Divergence evaluation: 50 harmful and 50 benign topic-matched prompts from the same sources, seeded with `random.Random(101)` and `random.Random(103)`.

Capability probes. REASON: MMLU `formal_logic` multiple-choice items; gold answer is the single letter A/B/C/D. ARITH: GSM8K-train problems, target is the numeric answer as a suffix. RECALL: TriviaQA-RC no-context questions, target is the gold answer span. INSTRUCT: FLAN-v2 prompts, target is the gold completion. Capability loss is the negative log-likelihood of the gold span conditional on the prompt. Sample sizes differ by measurement and are stated with each: the N gradient examples of the geometry (below), 64 per probe for the format ablation (Table 11) and Table 19, and the whole 126-item `formal_logic` split and 256 for the other probes in Table 20.

OLMo-2-0425-1B revision identifiers. Substrate sweep uses the HuggingFace branches `stage1-step0-tokens0B`, `stage1-step300-tokens1B`, `stage1-step10000-tokens21B`, `stage1-step30000-tokens63B`, `stage1-step50000-tokens105B`, `stage1-step200000-tokens420B`, `stage1-step500000-tokens1049B`, `stage1-step1000000-tokens2098B`, `stage1-step1907359-tokens4001B`. All are `stage1` branches (pretraining); we avoid the `stage2-ingredient*-*` branches (midtraining annealing), which would conflate data-mixture changes with pretraining-step progress.

Geometry measurement. $N = 256$ per-example gradients per (probe, block) for Pythia-1.4B and Qwen-2.5-7B, and $N = 128$ for Llama-3-8B and Mistral-7B-v0.3 to fit memory; $k = 128$ subspace rank. Per-layer MLP parameters only (the attention-head and layer-norm contributions to F_{cap} are omitted for budget and would at most add low-curvature directions that cannot meaningfully change the suppression/erasure classification).

Compute budget. All experiments ran on a 6 to 8× B200 node. Approximate per-experiment wall times: full geometry + benign-SFT attack + representation probe + divergence pipeline on a 7

to 8B Instruct model $\approx$ 30 to 60 min per pair (Qwen, Llama, Mistral); OLMo-2-1B substrate sweep across 9 pretraining revisions $\approx$ 45 min; per-revision SFT+DPO on OLMo-2-1B, two independent sets of 5 revisions (8 distinct; 20K SFT + 20K DPO $\times$2 epochs each) $\approx$ 3 hr per set; Pythia-410M from-scratch continuous-co-training run (10B tokens, 2-rank DDP) $\approx$ 14 hr; Pythia-160M continued-training pilot $\approx$ 1 hr; developmental-axis follow-up checks (multi-seed substrate sweep + cross-family gradient-norm) $\approx$ 30 to 45 min through a 6-GPU pool; from-scratch scale runs (Table 3), 410M and 1B on 2 GPUs each ($\approx$ 9 hr per run), 2.8B and 6.9B on 4 GPUs each ($\approx$ 22 to 28 hr and 40 to 46 hr per run). Total compute envelope for the experiments in this paper: $\sim$700 B200-GPU-hours, of which $\sim$610 are the scale runs; preliminary or failed experiments not reported add roughly 80 more.

Code and artifacts. All launcher scripts, per-experiment result bundles, and a single merged `combined_bundle.json` (9 experiments, 1.0 MiB) are in the anonymized repository.

B Geometry Instrument: Implementation Notes

Orthogonal decomposition of Δ (detail for Sec. 3). The overlap ratio in Sec. 3 uses $\Delta = \Delta_{\parallel} + \Delta_{\perp}$ with $\Delta_{\parallel} := U_k^{(c)}(U_k^{(c)})^{\top}\Delta$. Here $U_k^{(c)} \in \mathbb{R}^{d\times k}$ stacks the top-k eigenvectors of the empirical capability Fisher (equivalently, the top-k right singular vectors of the per-example gradient matrix G). Because $U_k^{(c)}$ has orthonormal columns, $U_k^{(c)}(U_k^{(c)})^{\top}$ is the orthogonal projector onto the top-k Fisher eigenspace and the decomposition is orthogonal: $\Delta_{\parallel} \perp \Delta_{\perp}$ and $\|\Delta\|_2^2 = \|\Delta_{\parallel}\|_2^2 + \|\Delta_{\perp}\|_2^2$ (Pythagoras). Orthonormality also gives $\|\Delta_{\parallel}\|_2^2 = \|(U_k^{(c)})^{\top}\Delta\|_2^2$, i.e., the squared norm of the projection equals the squared norm of its k-dimensional coefficient vector. We use this second form exclusively: the d-vector $\Delta_{\parallel}$ is never materialized, and $\rho^{(c)}$ reduces to a $(N,)$-sized intermediate (see the top-k eigenspace estimator below). $\Delta_{\perp}$ lives in directions of approximately zero Fisher curvature (outside the top-k eigenspace) and therefore does not contribute to first-order capability change, which is why only $\|\Delta_{\parallel}\|_2^2$ appears in $\rho^{(c)}$'s numerator.

Top-k eigenspace estimator. Our estimator of the top-k eigenspace of $F_{\text{cap}}^{(c)}$ from per-example gradients uses the N-side Gram trick. For N per-example gradients $G \in \mathbb{R}^{N\times d}$, the $d \times d$ Fisher is never formed; instead we compute the $N \times N$ Gram $GG^{\top}$, eigendecompose it, and recover the top-k right singular vectors of G as the capability eigenbasis. At $d \sim 2 \times 10^8$ (MLP parameter count for a 7B model) and $N \sim 128$, the full $fp32$ per-example gradient matrix $G \in \mathbb{R}^{N\times d}$ is $\sim$100 GiB (the $N \times N$ Gram $GG^{\top}$ itself is only $\sim$65 KB). We chunk the Fisher matmul along d in 4M-element slices and accumulate in $fp32$, bounding peak GPU memory at $\sim$2 GiB beyond the (N, d) gradient buffer. We never materialize the (d, k) capability-subspace matrix U_k explicitly: the projection $U_k^{\top}\Delta$ is computed directly from G, V_k, and singular values, which reduces to an $(N,)$-sized intermediate. This is what makes the pipeline run at Qwen/Llama scale on a single GPU.

Significance of the ρ alignment (random-subspace null). The claim $\rho \gg k/d$ is a statement about a null: a random k-subspace overlaps the capability eigenspace by k/d in expectation. We quantify how far the observed alignment sits from that null by drawing random k-subspaces at the true parameter dimensionality and forming a z-score, $z = (\rho_{\text{obs}} - \text{mean}_{\text{null}})/\text{std}_{\text{null}}$. Table 4 reports it for the three large models. The observed overlap is 21 to 67 times the k/d floor and 159 to 528 standard deviations above the random-subspace mean ($p \approx 0$, two-sided), so the alignment is small in absolute terms yet unambiguously non-random. The companion orthogonality statistic $|\cos(\nabla\mathcal{L}_{\text{cap}}, \Delta)| \approx 0.002$ sits about 150 standard deviations from a zero-mean null while still leaving Δ 99.8% orthogonal to the capability gradient; the lemma's conclusion, no first-order change in $\mathcal{L}_{\text{cap}}$, we check directly on $\Delta\mathcal{L}_{\text{cap}}$ (Sec. 4, App. F) rather than through this statistic.

Prompt format for the per-example gradients. All ρ and q_{norm} values reported in the paper use the raw 0-shot prompt convention described in Appendix F ("Question: ...\nAnswer:" with no chat template and no in-context demonstrations), the same format used for L_{cap}. Each per-example gradient is $\nabla_\theta \sum_t \log p_\theta(y_t \mid x, y_{<t})$ for the full gold span $y = (y_1, \ldots, y_T)$, a single token for REASON, multiple tokens for the other probes, evaluated at $\theta = W_{\text{base}}$. The top-k Fisher subspace therefore contains a mixture of (i) reasoning-relevant directions, which vary per example, and (ii) at

Table 4: Random-subspace significance of the Fisher alignment. ρ_{obs} is the observed overlap (max over layers); the null is a random k-subspace at the model's MLP parameter dimensionality d; $z = (\rho_{\text{obs}} - k/d)/\text{std}_{\text{null}}$. All $p \approx 0$ (two-sided). The alignment is 21 to 67 times the floor and hundreds of null standard deviations above it, so it is small but not a high-dimensional coincidence.

Model	ρ_{obs}	$\rho_{\text{obs}}/(k/d)$	z (SDs above null)
Qwen-2.5-7B	4.1×10^{-5}	$67\times$	528
Mistral-7B-v0.3	3.2×10^{-5}	$44\times$	343
Llama-3-8B	1.5×10^{-5}	$21\times$	159

most a handful of format-bias directions, which are shared across examples. With $k = 128$ and $N = 128$ to 256 examples, reasoning-relevant variance dominates: format-bias contributes at most a few of the k eigenvectors. Evaluating the Fisher at W_{base} (the completion-style base) rather than at W_{safe} further biases the top-k basis toward reasoning directions, because the base emits gold tokens natively and its log-likelihood's sharpest directions are content-related, not format-related.

3-shot robustness check. We re-ran the full geometry measurement (all three model families, all four probes, all 28 to 32 layers) with three in-context demonstrations prepended to each probe prompt, to test whether the ρ values are sensitive to prompt format. Table 5 reports mean ρ across layers and peak-q_{norm}-layer, for the raw 0-shot (paper default) and raw 3-shot conditions.

Primary claim (regime classification). Mean ρ is essentially unchanged between 0-shot and 3-shot: within $\pm 5\%$ for Qwen and Llama on every probe, within $\pm 20\%$ for Mistral. All three models stay at least two orders of magnitude below $\rho_{\text{part}} = 10^{-2}$ under 3-shot. The suppression classification does not depend on prompt format.

Secondary claim (q_{norm}-peak layer). Peak-q_{norm} layer is reproducible on the diagnostic probe (REASON) for Llama (L30 $\rightarrow$ L30) and Mistral (L12 $\rightarrow$ L12), confirming the Fig. 5 shaded bands. Qwen's REASON peak sits at block 0 in both conditions (525.6 at 0-shot), a first-block effect next to the embeddings where raw-gradient magnitude is large. The q_{norm} mass relevant to our claim is the mid-stack peak of Fig. 5 (220.6 at block 19), which is why Table 8 and every other peak-q_{norm} figure in the paper exclude block 0. Peak-q_{norm} layer for ARITH, RECALL, INSTRUCT is less stable across prompt formats, but these probes' peaks are not the basis of our safety-install-band argument.

The framework's prediction, reasoning variance dominates the top-k subspace regardless of in-context format, so ρ and the regime label carry over, is confirmed.

Table 5: Geometry robustness check under raw 3-shot probe prompts. Mean ρ averaged across all transformer blocks; q_{norm}-peak reports the layer where $q_{\text{norm}}{}^{(c)}$ attains its maximum. All three families remain in the suppression regime ($\rho \ll \rho_{\text{part}} = 10^{-2}$) under 3-shot; peak-$q_{\text{norm}}$ layer is stable on the probes driving the safety-install-band claim.

		mean ρ		q_{norm}-peak layer	
Model	Probe	0-shot	3-shot	0-shot	3-shot
Qwen-2.5-7B	REASON	1.40×10^{-5}	1.40×10^{-5}	L0	L0
	ARITH	1.21×10^{-5}	1.01×10^{-5}	L26	L0
	RECALL	1.20×10^{-5}	9.37×10^{-6}	L18	L27
	INSTRUCT	1.07×10^{-5}	1.05×10^{-5}	L27	L17
Llama-3-8B	REASON	5.07×10^{-6}	5.07×10^{-6}	L30	L30
	ARITH	6.08×10^{-6}	7.02×10^{-6}	L30	L30
	RECALL	5.81×10^{-6}	5.87×10^{-6}	L13	L31
	INSTRUCT	5.38×10^{-6}	5.41×10^{-6}	L31	L31
Mistral-7B-v0.3	REASON	1.21×10^{-5}	1.21×10^{-5}	L12	L12
	ARITH	1.34×10^{-5}	1.30×10^{-5}	L4	L29
	RECALL	9.42×10^{-6}	7.64×10^{-6}	L31	L0
	INSTRUCT	1.16×10^{-5}	1.12×10^{-5}	L3	L12

C Regime Classification Summary

Table 6 summarizes the dichotomy introduced prose-form in §3: *suppression* ($\rho \ll \rho_{\text{part}}$, Lemma 1 applies whenever the base is mature) vs. *erasure* ($\rho \gtrsim \rho_{\text{part}}$, Lemma fails; unobserved under DPO/RLHF; reachable by adversarial SFT, App. G; target of unlearning (Li et al., 2024)). Every DPO/RLHF observation in this paper lies in the suppression row: the largest block-mean ρ we measure is 1.1×10^{-4} (SFT+DPO at 1049B, Table 18), about $90\times$ below $\rho_{\text{part}} = 10^{-2}$, and the largest single-block value, 7.4×10^{-4} in that same install, is still $13\times$ below it. We identify two behaviorally-distinguishable sub-patterns *within* suppression, detailed in the right column.

Table 6: Regime dichotomy with two sub-patterns inside suppression. Every DPO/RLHF recipe we tested (Pythia-1.4B, Qwen-2.5-7B, Llama-3-8B, Mistral-7B-v0.3, OLMo-2-1B under Weak / Strong / SFT+DPO) lands in row 1; row 2 is unobserved under post-hoc alignment and is the target of unlearning (Li et al., 2024).

Regime	ρ	q_{norm}	Observed-in; sub-patterns within
Suppression	$\ll \rho_{\text{part}}$ (block means 10^{-6} to 1.1×10^{-4})	$\gg 1$ (thin and sharp) or ~ 1 (diffuse)	**Baseline** (pure DPO/RLHF on mature base, $\rho \sim 10^{-6}$ to 10^{-5}): Pythia-1.4B+DPO (§4); Qwen/Llama/Mistral-Instruct (§4); post-substrate OLMo Weak/Strong (§6). **Sub-pattern (a), under-trained base**: Lemma assumption fails, $q_{\text{norm}}^{\text{ARITH}} \in [600, 863]$ (pre-substrate OLMo step 300, §6). **Sub-pattern (b), SFT-stage recipe**: SFT's gradient lies in F_{cap}'s top-k, raising ρ 6 to $19\times$ over the Weak recipe to $\sim 10^{-4}$ (SFT+DPO at every post-substrate revision, §6).
Erasure	$\gtrsim \rho_{\text{part}}$	any	unobserved under DPO/RLHF; reachable by adversarial SFT on (prompt, wrong-answer) pairs (max $\rho \approx 10^{-2}$ on target probe; App. G); target of unlearning (RMU and related) (Li et al., 2024)

Behavioral implication per sub-pattern. The suppression baseline gives the "capability preserved, refusal fragile" asymmetry demonstrated in §4: the Qi benign-fine-tuning attack removes the thin gating circuit, and the geometry predicts it leaves the (off-subspace) capability representations in place, though we do not measure capability after the attack. Sub-pattern (a) means DPO can reduce $\mathcal{L}_{\text{cap}}$ at first order (Lemma assumption fails) and therefore installs a weaker, already-moving refusal; we observe $q_{\text{norm}}^{\text{ARITH}}$ up to 863 at pre-substrate OLMo step 300 and behaviorally, under SFT+DPO, a 21.9% pre-attack refusal that the attack collapses to 0%. Sub-pattern (b) installs a refusal whose in-subspace SFT component provides residual resistance: the per-revision run that keeps the most post-attack refusal (15.6%, against 3.1% at 105B and none elsewhere) is SFT+DPO at the final OLMo checkpoint. Erasure would decouple the capability from the model at leading order; the Lemma shows standard post-hoc alignment cannot reach it.

D Pythia-1.4B: Full Tables and Per-Layer Figure

Table 7: Pythia-1.4B post-DPO: per-probe ρ (mean over 24 transformer blocks), normalized Fisher-quadratic form q_{norm}, and capability-loss shift $\Delta\mathcal{L}_{\text{cap}}$. All rows are in the suppression regime; capability loss *decreases* slightly on every probe.

Probe	mean ρ_ℓ	mean $q_{\text{norm}\,\ell}$	$\Delta\mathcal{L}_{\text{cap}}$	Regime
REASON (MMLU fl)	1.7×10^{-5}	7.30	-0.015	suppression
ARITH (GSM8K)	2.4×10^{-5}	21.39	-0.027	suppression
RECALL (TriviaQA)	2.2×10^{-5}	10.73	-0.010	suppression
INSTRUCT (FLAN)	2.0×10^{-5}	6.50	-0.007	suppression

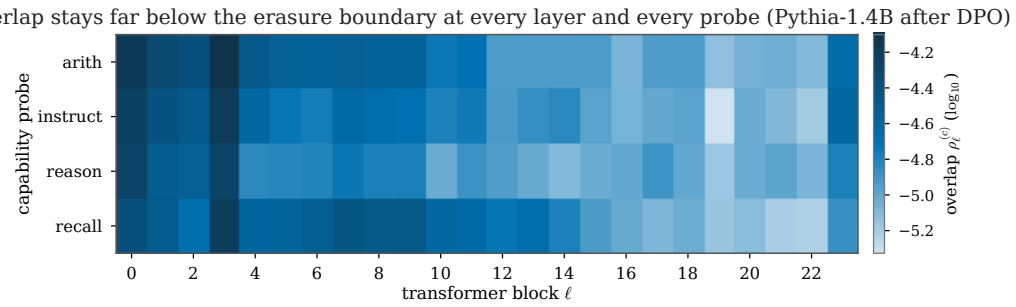


Figure 4: Per-layer ρ_ℓ (log-scale) for each capability probe on Pythia-1.4B after DPO. Layers 0, 3, and 23 (residual-stream boundaries and the final block) dominate; middle layers ($\sim$ 16 to 22) sit near the random floor. Pattern is consistent with per-layer localization reported for the refusal direction (Arditi et al., 2024).

Null behavior: the recipe installs no refusal. DPO on Pythia-1.4B step143000 with our recipe produces a pre-attack refusal rate of 0% on AdvBench, so there is no refusal for the attack to remove and attack-success numbers say nothing about the install. Applying the same topic-matched selective-divergence instrument as in Fig. 5 (50 harmful, 50 benign), $|\text{selective_div}_\ell| < 10^{-3}$ at every layer, with no monotone structure. For reference, the Qwen-2.5-7B-Instruct curve on the same instrument reaches $+0.166$. The Pythia null is therefore an internal consistency check: when no refusal circuit is installed behaviorally, the representation-shift instrument sees nothing. This is what motivated the move to OLMo-2-1B-Instruct (which ships with a Tulu-3-trained install that does produce refusal) for the substrate sweep (Sec. 6).

E Cross-Family Replication Details

Learned-classifier scoring. The same generations rescored by a harmful-output classifier are in App. L, which repeats both main-text attack tables under both detectors.

Table 8: Qwen-2.5-7B $\rightarrow$ Qwen-2.5-7B-Instruct: per-probe overlap ρ (mean / max over 28 blocks), normalized Fisher-quadratic form q_{norm}, raw-0-shot $\Delta\mathcal{L}_{\text{cap}}$ averaged over the 256 gradient examples. Max q_{norm} is over blocks 1 to 27: block 0, next to the embeddings, reaches 525.6 on REASON through raw-gradient magnitude alone and is excluded from every peak-q_{norm} figure in the paper except Table 5, which reports the unexcluded argmax (App. B). Source for the prose summary in §4.

Probe	mean ρ	max ρ	mean q_{norm}	max q_{norm} (layer)	$\Delta\mathcal{L}_{\text{cap}}$	Regime
REASON	1.4×10^{-5}	4.1×10^{-5}	76.2	220.6 (ℓ=19)	+3.45	suppression
ARITH	1.2×10^{-5}	2.9×10^{-5}	47.1	198.4 (ℓ=26)	+3.15	suppression
RECALL	1.2×10^{-5}	2.2×10^{-5}	29.8	94.7 (ℓ=18)	+1.07	suppression
INSTRUCT	1.1×10^{-5}	2.1×10^{-5}	18.6	31.3 (ℓ=27)	+0.67	suppression

Llama-3-8B. We use Llama-3-8B as W_{base} and Llama-3-8B-Instruct as W_{safe}. ρ runs from 2.8×10^{-7} to 1.5×10^{-5} across 32 blocks and four probes, up to 21$\times$ the random floor $k/d \approx 7 \times 10^{-7}$; four early-block cells sit below it. q_{norm} peaks for REASON and ARITH at blocks 30 to 31 (48.9 and 47.7); much later than Qwen's mid-block peak. The 100-example benign Alpaca fine-tune strips nearly all of its refusal (Table 1; compliance NLL 6.18 $\rightarrow$ 4.30). Selective divergence peaks at $+0.053$ at block 12, roughly 3$\times$ smaller than Qwen's $+0.166$, with late layers going slightly negative (-0.02), reflecting that only 49% of harmful prompts actually engage the refusal circuit, so harm-vs-benign divergence is half-dominated by chat-style effects rather than refusal-specific ones. Raw-0-shot $\Delta\mathcal{L}_{\text{cap}}$ is small on Llama (REASON + 0.20, ARITH + 0.88), well under the $+1.1$ 3-shot-neutral ceiling; Llama-3-Instruct is less chat-template-dependent than Qwen.

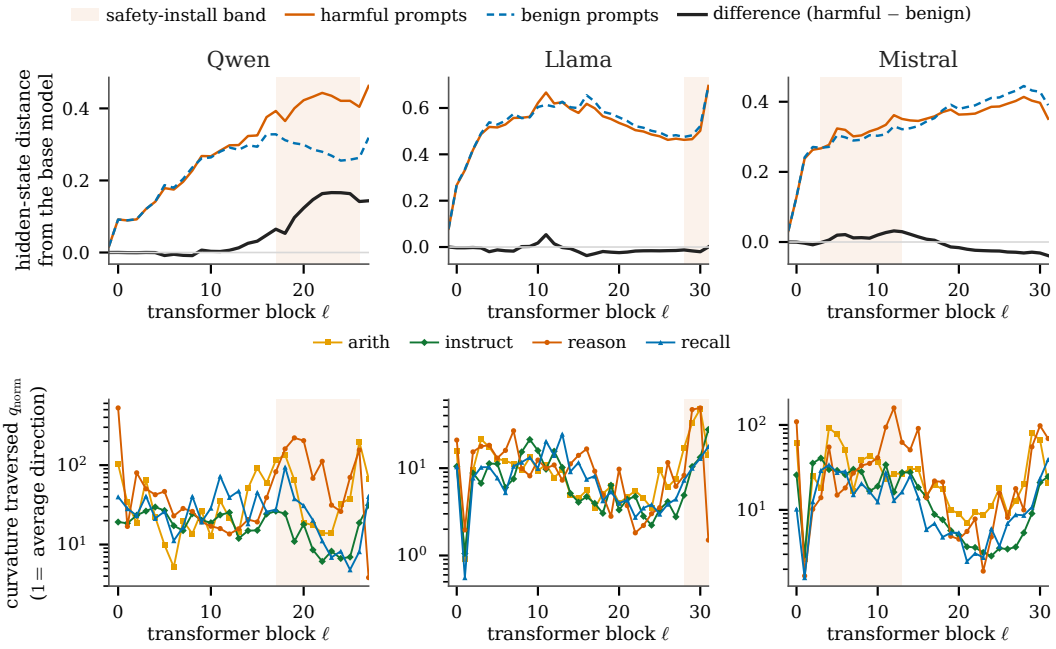


Figure 5: Layer-localized signals across three model families (columns: Qwen-2.5-7B-Instruct, Llama-3-8B-Instruct, Mistral-7B-Instruct-v0.3), referenced from Sec. 4. The top row is Fig. 2 in the main text, repeated here so the two rows can be read against each other. **Top row:** per-layer L2 divergence between base and instruct residual streams on harmful vs. benign topic-matched prompts (n_{pairs}=50 each), plus selective divergence (harm − benign). **Bottom row:** per-layer ${q_{\text{norm}}}^{(c)}$ for the four capability probes (log-scale). Shaded bands bracket each model's ${q_{\text{norm}}}^{\text{REASON}}$/${q_{\text{norm}}}^{\text{ARITH}}$ peaks (Qwen ℓ=17 to 26; Llama ℓ=28 to 31; Mistral ℓ=3 to 13). Selective-divergence magnitude scales with pre-attack refusal rate: +0.166 (Qwen, 93%), +0.053 (Llama, 49%), +0.032 (Mistral, 1.2%).

Table 9: Cross-model summary across three instruct-tuned 7 to 8B models. All replicate the suppression regime; the absolute q_{norm} band is family-specific; the behavioral attack number is a joint measurement of (refusal installed) $\times$ (robustness of that install). Peak q_{norm} excludes Qwen's block 0 (Table 8). Geometry only; the behavioral effect of the attack is Table 1 in the main text, measured in a single run under both detectors.

Signal	Qwen-2.5-7B-Instruct	Llama-3-8B-Instruct	Mistral-7B-Instruct-v0.3
mean ρ	1.2×10^{-5}	5.6×10^{-6}	1.2×10^{-5}
peak $\rho/(k/d)$	$67\times$	$21\times$	$44\times$
peak $q_{\text{norm}}^{\text{REASON}}$ (layer)	220.6 (ℓ=19)	48.9 (ℓ=30)	158.7 (ℓ=12)
peak $q_{\text{norm}}^{\text{ARITH}}$ (layer)	198.4 (ℓ=26)	47.7 (ℓ=30)	92.1 (ℓ=4)
peak selective divergence	+0.166 (ℓ=24)	+0.053 (ℓ=12)	+0.032 (ℓ=13)
max raw-0-shot $\Delta\mathcal{L}_{\text{cap}}$	+3.45	+0.88	+0.59

Mistral-7B-Instruct-v0.3. ρ runs from 1.1×10^{-6} to 3.2×10^{-5}, 1.6 to $44\times$ the random floor $k/d \approx 7 \times 10^{-7}$, mean $\rho = 1.2 \times 10^{-5}$. Peak-$q_{\text{norm}}$ band is early-to-mid residual: $q_{\text{norm}}^{\text{REASON}}$ peaks at 158.7 at $\ell = 12$; $q_{\text{norm}}^{\text{ARITH}}$ at 92.1 at $\ell = 4$ (see Fig. 5, right column). All three model families land in the suppression regime, but the absolute layer of the sharp q_{norm} band is family-specific. Behaviorally, Mistral-7B-Instruct-v0.3 is an outlier: 1.2% pre-attack refusal on AdvBench, so the Qi benign-fine-tuning attack has almost nothing to break. Post-attack refusal is 0.4% (post-attack ASR $= 99.6\%$ trivially). Compliance NLL *rises* from 5.01 to 6.13: benign Alpaca SFT pushes Mistral-Instruct further from a naked "Sure, here is" continuation. With almost no refusal to remove, Mistral's attack numbers say little about fragility.

What the base already refuses. $\Delta = W_{\text{safe}} - W_{\text{base}}$ reads as a safety *install* only to the extent that W_{base} refuses nothing, so we measure the three shipped bases directly under the same prompts and detectors as their Instruct counterparts (no chat template is applied to either, so the pair is matched). Llama and Mistral behave as assumed: at most 1.6% by regex, and 7.0 to 19.9% by the classifier, which counts any completion it does not flag as harmful as a refusal. Qwen does not: its base already refuses 71.1% of AdvBench by regex and 69.5% by classifier, so its shipped Δ raises AdvBench refusal by about 22 pp rather than creating it. This does not affect the geometry, which characterizes whatever Δ is, but it does mean the Qwen pair should be read as a partial install, and it is why the safety-only controls of Sec. 5 carry the weight they do. It also sharpens the fragility result: the attack leaves Qwen-Instruct at 58.2% on AdvBench (Table 1), *below* the 71.1% its own base refuses, so fine-tuning on 100 benign examples does not merely undo the install but leaves the model more compliant than the checkpoint it was built from.

Table 10: Refusal (%) of the three shipped *base* models, before any install, under both detectors. Same prompts, decoding and scoring as Table 1, with no chat template applied to either half of a pair. $n = 256/200/256$.

	AdvBench		HarmBench		StrongReject	
Base model	regex	classifier	regex	classifier	regex	classifier
Qwen-2.5-7B	**71.1**	**69.5**	31.5	34.0	28.5	30.9
Llama-3-8B	0.0	7.0	1.0	10.5	1.6	12.5
Mistral-7B-v0.3	0.0	16.4	0.0	14.0	0.8	19.9

F Chat-Template Ablation: Full Four-Condition Table

Scoring convention. Each probe defines a gold span that we score the NLL of, conditional on the prompt. For REASON (MMLU `formal_logic`) the gold span is a single letter (`" A"`, `" B"`, `" C"`, or `" D"`). For ARITH (GSM8K) it is the numeric answer as a suffix. For RECALL (TriviaQA) it is the answer span. For INSTRUCT (FLAN-v2) it is the gold completion. This is the same NLL-of-gold-span convention used by `lm-evaluation-harness` and HELM for MMLU-style benchmarks; the *scoring* is standard.

Why $\Delta\mathcal{L}_{\text{cap}}$ can inflate across a base/instruct pair. What is not automatically standard is using L_{cap} as a *capability-preservation diagnostic across two differently-aligned models*. The value L_{cap} is sensitive to how each model emits the first post-prompt tokens. A base model trained on raw-completion corpora tends to continue `"...Answer:"` with the bare letter directly, since that matches its training distribution. An instruct model trained with chat-style responses often continues with `"The answer is A, because..."` or an explanatory preamble that puts a newline or word before the gold letter. The probability mass on `" A"` drops, L_{cap} rises, and ΔL_{cap} looks large. The underlying knowledge is unchanged; only the output distribution's *format* has shifted.

Ablation design. We run each probe under four prompt formats on Qwen-2.5-7B and Qwen-2.5-7B-Instruct. *raw (0-shot)*: the prompt ends at "Answer:" and we score the gold span directly. *raw (3-shot)*: three worked examples precede the prompt (in Q/A pairs), then the query, following a standard few-shot convention. *chat (0-shot)*: the prompt is wrapped in Qwen's chat template. *chat (3-shot)*: chat template plus three in-context demonstrations. Table 11 reports $\Delta L_{\text{cap}} = L_{\text{cap}}(W_{\text{safe}}) - L_{\text{cap}}(W_{\text{base}})$ per condition.

Table 11: Chat-template ablation on Qwen-2.5-7B. Four prompt formats on the same four capability probes, $n = 64$ examples per probe, three demonstrations in the 3-shot columns. The raw 0-shot column uses a different sample from Table 8, which averages over the 256 gradient examples, so the two differ by up to 0.21 nats (INSTRUCT). 3-shot collapses the gap by 40 to 74% on REASON, ARITH and RECALL, and by 20% on INSTRUCT, whose raw gap is already small, by showing the Instruct model the raw-completion format via in-context examples. Chat formatting *inflates* the gap because we score NLL of a bare gold token that Instruct no longer naturally emits after its chat template; 3-shot partially recovers by demonstrating the target format in context.

Probe	raw (0-shot)	raw (3-shot)	chat (0-shot)	chat (3-shot)
REASON	+3.35	+0.87	+21.01	+11.07
ARITH	+3.06	+1.10	+15.19	+9.14
RECALL	+0.90	+0.54	+5.39	+0.84
INSTRUCT	+0.46	+0.37	+3.16	+2.99

Reading the table. The raw-0-shot column (first data column) reproduces the ordering and approximate size of the $\Delta\mathcal{L}_{\text{cap}}$ values in Table 8, on a smaller and different sample. The raw-3-shot column shows the gap collapse to $+0.37$ to $+1.10$ nats across probes; this is our "format-neutralized" estimate of the true capability-loss shift and is small enough to be consistent with the geometric claim $\rho \ll \rho_{\text{part}}$. The chat columns go in the opposite direction: with the Instruct model's expected chat template applied, the NLL of the bare gold token *rises* further, because the Instruct model even more strongly prefers to emit a preamble under its native format. This inversion indicates that the raw-0-shot $\Delta\mathcal{L}_{\text{cap}}$ in Table 8 is largely a format-shift effect rather than a capability shift.

G ERASURE IS REACHABLE: ADVERSARIAL-SFT DEMONSTRATION

None of the safety-training recipes studied in the main body crosses ρ_{part}. To establish that the erasure regime is not a definition without referents, we run a stress test that engineers an update which should, by construction, land inside F_{cap}'s top-k eigenspace: supervised fine-tuning on (prompt, wrong-answer) pairs drawn from the capability probe itself. The gradient of NLL on a wrong gold token lies in the same directions the probe's Fisher picks out, so in-subspace mass should accumulate with training steps. This is a deliberately contrived recipe and not a realistic alignment procedure; its purpose is a boundary-reachability check.

Setup. Base model: Pythia-1.4B `step143000`, matching §4. Two target probes are studied in parallel, one per independent adversarial-SFT run:

- *Target*=REASON: MMLU `formal_logic`, gold letter replaced by a uniformly-sampled non-gold letter. 2000 pairs, 2 epochs, LR 2×10^{-5}, AdamW, bf16.
- *Target*=ARITH: GSM8K training split, numeric gold answer perturbed by a small random offset preserving magnitude. Same 2000/2/2e$-$5 schedule.

The geometry instrument of §3 is then applied to each resulting W_{safe} against the Pythia-1.4B base: $N = 128$ per-example gradients, $k = 128$ per-layer subspace, evaluated on all four capability probes (target and three off-target).

Result: target-probe ρ reaches the ρ_{part} boundary. Table 12 summarizes. On the target probe, max ρ reaches 1.04×10^{-2} (target=REASON), at ρ_{part}, and 3.01×10^{-3} (target=ARITH). Mean ρ on target is 3.33×10^{-3} and 1.16×10^{-3} respectively, 10 to 30× the SFT+DPO sub-pattern (b) values in §6. q_{norm} on target reaches 33,945 (reason) and 15,064 (arith), more than an order of magnitude above the pre-substrate OLMo ${q_{\text{norm}}}^{\text{ARITH}}$ of 863. The $\rho_{\text{part}} = 10^{-2}$ threshold is thus empirically anchored: it lies exactly where an update built to saturate F_{cap}'s top-k in fact reaches.

target probe	probe under eval	mean ρ	max ρ	max q_{norm}	$\Delta\mathcal{L}_{\text{cap}}$	target?
REASON	REASON	3.33×10^{-3}	$\mathbf{1.04 \times 10^{-2}}$	33945	-0.96	✓
REASON	ARITH	1.16×10^{-3}	4.66×10^{-3}	15773	-0.62	
REASON	RECALL	4.97×10^{-4}	1.49×10^{-3}	1895	-0.35	
REASON	INSTRUCT	4.12×10^{-4}	1.33×10^{-3}	328	$+0.15$	
ARITH	ARITH	1.16×10^{-3}	3.01×10^{-3}	15064	-2.08	✓
ARITH	REASON	3.47×10^{-4}	8.95×10^{-4}	2956	-0.72	
ARITH	RECALL	2.78×10^{-4}	8.28×10^{-4}	1151	-0.38	
ARITH	INSTRUCT	1.84×10^{-4}	8.18×10^{-4}	316	$+0.23$	

Table 12: Adversarial-SFT erasure demonstration on Pythia-1.4B `step143000`. One SFT run per target probe; geometry measured against the base on all four probes. Max ρ on the target probe reaches the $\rho_{\text{part}} = 10^{-2}$ boundary for target=REASON and comes within about 3× of it for target=ARITH; q_{norm} on target is 17 to 39× the largest value seen under DPO/RLHF (863, pre-substrate OLMo). Off-target probes see 3 to 8× lower mean ρ than the target, establishing target specificity. $\Delta\mathcal{L}_{\text{cap}}$ is NLL-based; on capability probes with a stable wrong-token format (REASON, ARITH), adversarial SFT *lowers* NLL because format-learning dominates at this intervention scale (see caveat below). $\rho_{\text{rand}} = 3.75 \times 10^{-6}$ throughout.

Target specificity. Off-target probes see material but consistently lower ρ than the target probe: target=REASON elevates ARITH to 1.16×10^{-3} (2.9× below target REASON) and RECALL/INSTRUCT near 4 to 5×10^{-4} (6 to 8× below). Target=ARITH elevates REASON to 3.47×10^{-4}, RECALL to 2.78×10^{-4}. The ratio $\rho^{\text{target}}/\rho^{\text{off-target}}$ is stable across both runs, supporting a probe-local interpretation of F_{cap}'s top-k: the eigendirections carrying reasoning and arithmetic overlap but are not identical, and an update engineered for one is not trivially an update for the other. This matters for downstream unlearning: an adversarial-SFT-style erasure on one skill is not a proxy for blanket capability removal.

NLL-vs-argmax caveat. The $\Delta\mathcal{L}_{\text{cap}}$ column shows that NLL on the target probe *decreases* after adversarial SFT (-0.96 on REASON, -2.08 on ARITH), the opposite of what a literal capability-loss reading would suggest. This is the format-learning effect: a Pythia-1.4B base emits letter and digit tokens after "Answer:" with nontrivial entropy, and 2000 demonstrations of *any* single-token answer (correct or not) sharpen the format distribution enough to lower NLL on the bare gold token. The intervention sharpens the format while corrupting content; argmax accuracy would drop where NLL improves. For consistency with the geometry machinery elsewhere in the paper we report NLL here, but the behavioral interpretation should be: adversarial SFT installs a large in-subspace perturbation ($\rho \to \rho_{\text{part}}$), the lemma's erasure prediction applies (first-order capability change is permitted; the sign depends on what was optimized), and the NLL metric becomes unreliable because it does not factor format from content. A cleaner probe for future work is top-1 accuracy on held-out target-probe questions.

Interpretation. The result closes the boundary-reachability question without claiming that adversarial SFT is a practical unlearning method: it is not; it is a deliberately over-specified recipe designed to populate the erasure regime empirically. The takeaway for the main text is structural: $\rho_{\text{part}} = 10^{-2}$ is not an arbitrary threshold but the value that an update built to saturate F_{cap}'s top-k eigenspace at this scale in fact reaches, and every suppression observation in the paper sits at

least $90\times$ below it as a block mean, and $13\times$ below it at its peak-ρ block. The qualitative regime separation between suppression and erasure is therefore well-anchored on both sides.

H Full Proof of Lemma 1 (Kernel Immobility)

We restate the lemma and give a full proof.

Lemma 1 (Kernel immobility, restated). *Let $\mathcal{L}_{cap}^{(c)} : \mathbb{R}^d \to \mathbb{R}$ be a capability loss, twice continuously differentiable on a neighborhood $\mathcal{N}$ of W_{base}, with Lipschitz Hessian on $\mathcal{N}$. Let*

$$F_{cap}^{(c)} := \mathbb{E}_{(x,y)\sim\mathcal{D}_c}\left[\nabla_\theta \log p_\theta(y|x)\, \nabla_\theta \log p_\theta(y|x)^\top\right]\Big|_{\theta=W_{base}}$$

be the empirical Fisher information matrix. Let $E := H(W_{base}) - F_{cap}^{(c)}$ denote the Gauss-Newton residual. If $\Delta \in \ker(F_{cap}^{(c)})$, then

$$\mathcal{L}_{cap}^{(c)}(W_{base}+\Delta) - \mathcal{L}_{cap}^{(c)}(W_{base}) \;=\; \tfrac{1}{2}\Delta^\top E\,\Delta \;+\; O(\|\Delta\|_2^3).$$

Under exact Gauss-Newton ($E = 0$, e.g., at a well-specified mature checkpoint where $p_\theta(\cdot \mid W_{base})$ matches $\mathcal{D}_c$ (Martens, 2020)), this strengthens to $O(\|\Delta\|_2^3)$. Either way, no first-order post-hoc update can change $\mathcal{L}_{cap}^{(c)}$ at leading order.

Proof. **Step 1 (Taylor expansion).** By Taylor's theorem with Lagrange remainder, for any Δ with $W_{\text{base}} + \Delta \in \mathcal{N}$,

$$\mathcal{L}_{\text{cap}}^{(c)}(W_{\text{base}}+\Delta) = \mathcal{L}_{\text{cap}}^{(c)}(W_{\text{base}}) + \nabla\mathcal{L}_{\text{cap}}^{(c)}(W_{\text{base}})^\top\Delta + \frac{1}{2}\Delta^\top H(W_{\text{base}})\Delta + R(\Delta),$$

where $H(W_{\text{base}}) = \nabla^2\mathcal{L}_{\text{cap}}^{(c)}(W_{\text{base}})$ is the Hessian and the remainder satisfies $|R(\Delta)| \le \frac{L}{6}\|\Delta\|_2^3$ by the Lipschitz-Hessian assumption.

Step 2 (linear term vanishes from the kernel condition). The kernel hypothesis $\Delta \in \ker(F_{\text{cap}}^{(c)})$ gives $\Delta^\top F_{\text{cap}}^{(c)}\Delta = \mathbb{E}_{(x,y)\sim\mathcal{D}_c}\left[(\Delta^\top\nabla\log p_\theta(y|x))^2\right] = 0$. As an expectation of a nonnegative quantity, this forces $\Delta^\top\nabla\log p_\theta(y|x) = 0$ for $\mathcal{D}_c$-almost every (x, y). Taking expectations again,

$$\Delta^\top\nabla\mathcal{L}_{\text{cap}}^{(c)}(W_{\text{base}}) \;=\; -\,\Delta^\top\mathbb{E}_{(x,y)\sim\mathcal{D}_c}[\nabla\log p_\theta(y|x)] \;=\; 0.$$

The linear Taylor term thus vanishes from the kernel condition alone, irrespective of whether W_{base} is a stationary point of $\mathcal{L}_{\text{cap}}^{(c)}$.

Step 3 (Hessian, Fisher, and the Gauss-Newton residual). For the likelihood loss $\mathcal{L}_{\text{cap}}^{(c)} = -\mathbb{E}_{(x,y)\sim\mathcal{D}_c}[\log p_\theta(y|x)]$, the calculus identity $\nabla^2\log p = \nabla^2 p/p - \nabla\log p(\nabla\log p)^\top$ gives

$$H(W_{\text{base}}) \;=\; -\,\mathbb{E}\big[\nabla^2\log p\big] \;=\; \mathbb{E}\big[\nabla\log p\,\nabla\log p^\top\big] \;-\; \mathbb{E}\big[\nabla^2 p/p\big] \;=\; F_{\text{cap}}^{(c)} \;-\; \mathbb{E}\big[\nabla^2 p/p\big]\,.$$

Bartlett's second identity gives $\mathbb{E}_{p_\theta}[\nabla^2 p/p] = \int\nabla^2 p\,\mathrm{d}y = \nabla^2\int p\,\mathrm{d}y = 0$ when expectations are taken under the model's own output distribution. When $p_\theta(\cdot \mid W_{\text{base}})$ is well-specified for $\mathcal{D}_c$ (which holds at a well-trained mature checkpoint), the Gauss-Newton residual $E := H(W_{\text{base}}) - F_{\text{cap}}^{(c)} = -\mathbb{E}_{\mathcal{D}_c}[\nabla^2 p/p]$ vanishes; this is the standard Gauss-Newton approximation (Martens, 2020). In general E is a finite matrix.

Step 4 (combining). Substituting Step 2 into Step 1 and decomposing $H = F_{\text{cap}}^{(c)} + E$,

$$\mathcal{L}_{\text{cap}}^{(c)}(W_{\text{base}}+\Delta) - \mathcal{L}_{\text{cap}}^{(c)}(W_{\text{base}}) \;=\; \tfrac{1}{2}\Delta^\top F_{\text{cap}}^{(c)}\Delta \;+\; \tfrac{1}{2}\Delta^\top E\,\Delta \;+\; R(\Delta).$$

The kernel hypothesis kills the first term ($\Delta^\top F_{\text{cap}}^{(c)}\Delta = 0$). The Lipschitz-Hessian assumption gives $|R(\Delta)| \le \frac{L}{6}\|\Delta\|_2^3$, so

$$\mathcal{L}_{\text{cap}}^{(c)}(W_{\text{base}}+\Delta) - \mathcal{L}_{\text{cap}}^{(c)}(W_{\text{base}}) \;=\; \tfrac{1}{2}\Delta^\top E\,\Delta \;+\; O(\|\Delta\|_2^3) \;=\; O(\|E\|_2\cdot\|\Delta\|_2^2) \;+\; O(\|\Delta\|_2^3).$$

Under exact Gauss-Newton ($E = 0$), this collapses to $O(\|\Delta\|_2^3)$.

Step 5 (first-order corollary). A first-order optimizer (SGD, Adam, AdamW) minimizing an auxiliary objective $\mathcal{L}_{\text{safe}}$ produces an update of the form $\Delta = -\eta\, M \nabla \mathcal{L}_{\text{safe}} + O(\eta^2)$, where M is a (possibly diagonal, possibly identity) preconditioner. The lemma's premise is on Δ itself, not on $\nabla \mathcal{L}_{\text{safe}}$ directly, so preconditioning does not weaken the conclusion: the lemma applies whenever the resulting Δ satisfies $\Delta \in \ker(F_{\text{cap}}^{(c)})$, which is the condition we measure empirically through $\rho^{(c)} = \|\Delta_{\|}\|_2^2 / \|\Delta\|_2^2$. □

Remark. The kernel condition $\Delta \in \ker(F_{\text{cap}}^{(c)})$ does the work in Step 2 alone: the linear Taylor term vanishes regardless of whether W_{base} is a stationary point of $\mathcal{L}_{\text{cap}}^{(c)}$. The local-minimum / well-specification assumption enters only at Step 3, where it underpins the Gauss-Newton approximation $H(W_{\text{base}}) \approx F_{\text{cap}}^{(c)}$. Under-trained checkpoints (e.g., the pre-substrate revisions in Sec. 6) violate well-specification: $\|E\|$ is large there, the strong $O(\|\Delta\|^3)$ form fails, and even an in-kernel update can shift $\mathcal{L}_{\text{cap}}$ at $O(\|\Delta\|^2)$. The update measured there is also far from the kernel ($q_{\text{norm}}{}^{\text{ARITH}}$ up to 863), so the linear term can act as well. Both routes are consistent with the under-trained-base sub-pattern, in which safety training still moves $\mathcal{L}_{\text{cap}}$ directly. The post-substrate mature checkpoint where gradient optimization has converged is where $\|E\| \approx 0$ and the cubic bound applies; this is the regime the paper's headline claim sits in.

I Developmental-Axis Follow-Up: Alignment, Cross-Family Grad-Norm, Plateau Noise

We report three follow-up checks tied to claims in §6 and §5: (i) the empirical alignment $\cos(\nabla \mathcal{L}_{\text{cap}}, \Delta)$ for a second, independent set of SFT+DPO updates at five OLMo-2-1B revisions; (ii) the gradient-norm instrument applied to Pythia-1.4B revisions (cross-family lemma sanity check); (iii) a multi-seed re-run of the substrate-divergence sweep to bound plateau noise. Table 13 first gives the full per-revision substrate sweep behind Fig. 3.

Table 13: Substrate sweep on OLMo-2-1B (full data behind Fig. 3): W_{base} swept across nine pretraining revisions spanning 0 to 4T tokens, W_{safe} = OLMo-2-1B-Instruct fixed. The random floor for this model is $k/d \approx 2.6 \times 10^{-6}$. Selective divergence goes from noise at 1B to plateau by 63B pretraining tokens, with the first signal at 21B. $\ell^\star$ = the residual-stream position (stream after $\ell^\star$ blocks) with the largest selective divergence; reported as n/a at pre-substrate checkpoints where the curve is flat at noise ($\sim 10^{-4}$) and argmax would label a random fluctuation rather than a locus.

Step	Tokens	mean ρ	max selective_div$_\ell$	$\ell^\star$
0	0B	1.4×10^{-6}	$+4 \times 10^{-4}$	n/a
300	1B	1.6×10^{-6}	$+4 \times 10^{-4}$	n/a
1×10^4	21B	1.8×10^{-6}	$+\mathbf{2.7 \times 10^{-2}}$	16
3×10^4	63B	1.3×10^{-6}	$+\mathbf{7.3 \times 10^{-2}}$	16
5×10^4	105B	1.0×10^{-6}	$+7.7 \times 10^{-2}$	6
2×10^5	420B	9.3×10^{-7}	$+7.8 \times 10^{-2}$	9
5×10^5	1049B	9.1×10^{-7}	$+4.9 \times 10^{-2}$	9
1×10^6	2098B	8.3×10^{-7}	$+5.9 \times 10^{-2}$	9
1.9×10^6	4001B	3.4×10^{-6}	$+7.1 \times 10^{-2}$	11

Gradient-orthogonality at the OLMo developmental axis. Table 14 reports $\cos(\nabla \mathcal{L}_{\text{cap}}^{(c)}(W_{\text{base}}), \Delta)$ averaged across the four probes, where $\Delta = W_{\text{safe}} - W_{\text{base}}$ is the SFT+DPO update at each revision, from the second run set described in the table caption. The relevant first-order capability-loss change per unit $\|\Delta\|^2$ is the linear coefficient $(\nabla \mathcal{L}_{\text{cap}})^\top \Delta / \|\Delta\|^2$.

Safety-only controls (Table 15). The shipped deltas fold instruction-following and safety together, so Sec. 5 isolates safety with two updates trained on a fixed Qwen-2.5-1.5B base with no general instruction-tuning: a refusal-SFT-only update and a DPO-only update. Every cosine is below 2×10^{-4} in magnitude, on both recipes and all four probes, although the two updates differ in size by a factor of 450.

Table 14: Alignment of the safety update with the capability gradient at OLMo developmental-axis revisions. Mean is over the four capability probes; Δ is an SFT+DPO update with the recipe of App. J, from a second, independent run set placed at 21B and 42B to resolve the transition, which shares only the 1B, 105B and 4001B revisions with the runs of Table 18. Across the entire axis $|\cos(\nabla\mathcal{L}_{\text{cap}}, \Delta)| < 0.002$, including the over-trained 4001B-token endpoint where $\|\nabla\mathcal{L}_{\text{cap}}\|$ peaks at 29.6. Translated into absolute nats, the linear Taylor term $(\nabla\mathcal{L}_{\text{cap}})^\top\Delta = \cos\cdot\|\nabla\mathcal{L}_{\text{cap}}\|\cdot\|\Delta\|$ is at most ~ 0.11 nats at the pre-substrate step300 revision and ≤ 0.06 nats at post-substrate mature revisions: a fraction of 10^{-3} of the maximum-aligned upper bound $\|\nabla\mathcal{L}_{\text{cap}}\|\cdot\|\Delta\| \in [71, 260]$, confirming the kernel-induced gradient orthogonality empirically.

Revision	$\|\Delta\|$	mean $\|\nabla\mathcal{L}_{\text{cap}}\|$	mean $\cos(\nabla\mathcal{L}_{\text{cap}}, \Delta)$	mean linear/$\|\Delta\|^2$
step300-tokens1B	12.69	20.46	-6×10^{-4}	-7×10^{-4}
step10000-tokens21B	9.65	13.69	-5×10^{-4}	-6×10^{-4}
step20000-tokens42B	9.11	10.09	-1×10^{-4}	-3×10^{-5}
step50000-tokens105B	8.87	7.97	0 to 4 d.p.	$+3 \times 10^{-4}$
step1907359-tokens4001B	8.12	29.56	-4×10^{-4}	-6×10^{-4}

Table 15: Safety-only controls on Qwen-2.5-1.5B: $\cos(\nabla\mathcal{L}_{\text{cap}}, \Delta)$ per capability probe for a refusal-SFT-only and a DPO-only update, $N = 128$ per-example gradients. Neither update contains general instruction-tuning, so the orthogonality cannot come from it.

Update	$\|\Delta\|$	REASON	RECALL	ARITH	INSTRUCT
refusal-SFT only	6.99	-1.1×10^{-4}	$+1.4 \times 10^{-5}$	$+1.7 \times 10^{-4}$	-6.0×10^{-5}
DPO only	0.0155	-1.6×10^{-4}	$+9.5 \times 10^{-5}$	$+6.1 \times 10^{-5}$	-4.8×10^{-5}

Pythia-1.4B grad-norm cross-family replication. The U-shape in $\|\nabla\mathcal{L}_{\text{cap}}\|$ across OLMo-2-1B (Table 14: large at 1B and 4T tokens, smallest at 105B) might have been specific to OLMo-2-1B's $\sim 200\times$-Chinchilla overtraining ratio. We re-apply the gradient-norm instrument across eight Pythia-1.4B pretraining revisions (Table 16). *The U-shape replicates*: $\|\nabla\mathcal{L}_{\text{cap}}\|$ falls from 119 at `step512` to a minimum of 28.8 at `step30000` ($\sim$ 63B tokens, $\sim$ 45 tokens/param, within about a factor of two of the compute-optimal ratio of $\sim$ 20), then rises again as Pythia is trained past compute-optimal, reaching 50.5 at the final `step143000` ($\sim$ 300B tokens, $\sim$ 210 tokens/param, $\sim 10\times$ Chinchilla). This argues against an OLMo-specific overtraining artifact; the U-shape looks like a generic property of pretraining, for which weight-decay-driven non-stationarity on either side of the compute-optimal point is one possible cause.

Table 16: $\|\nabla\mathcal{L}_{\text{cap}}\|$ across Pythia-1.4B pretraining revisions, mean over the four capability probes. The U-shape from OLMo replicates: minimum at `step30000` ($\sim$ 63B tokens, about twice the compute-optimal ratio), large at both extremes.

Revision	Tokens	mean $\|\nabla\mathcal{L}_{\text{cap}}\|$	REASON	ARITH	RECALL	INSTRUCT
`step512`	1B	**119.5**	146.7	209.9	98.7	22.5
`step1000`	2B	96.0	100.5	140.3	128.5	14.9
`step3000`	6B	38.8	51.5	72.2	22.3	9.1
`step10000`	21B	32.6	50.4	51.5	21.0	7.5
`step30000`	63B	**28.8**	31.2	56.1	19.8	8.2
`step60000`	126B	32.7	41.4	59.3	22.4	7.8
`step100000`	210B	41.8	51.0	75.1	30.5	10.5
`step143000`	300B	**50.5**	63.0	91.4	35.0	12.8

Substrate-plateau noise floor. The single-seed substrate sweep (§6) showed apparent within-plateau bouncing ($\max$ selective_div$_\ell \in [0.04, 0.12]$ across 42 to 210B; argmax layer $\in \{6, 7, 8, 9, 16\}$) at a single deterministic prompt order (n_{pairs}=50 from a pool of 60). We re-run with three shuffled seeds per revision (Table 17). Within-revision standard deviation is 0.002 to 0.010, an order of magnitude smaller than the between-revision spread of ~ 0.08. Argmax layer is

stable across seeds (one-position drift only at 168B, 210B). The two peaks at 84B and 147B sit at least 2.5 standard deviations above each neighbor, so they are real structure, not noise. The corrected reading: substrate engagement begins at $\sim$42B, reaches a first peak at 84B, dips moderately at 126B, peaks again at 147B, and stabilizes at $\sim$0.07 from 168B onward.

Table 17: Substrate divergence at 3 shuffled-seed runs per revision (mean $\pm$ std). Argmax layers reported per-seed, in run order.

Revision	$\max$ selective_div$_\ell$	argmax layer (per seed)
`step20000-tokens42B`	0.043 ± 0.002	16, 16, 16
`step40000-tokens84B`	0.100 ± 0.008	9, 9, 9
`step60000-tokens126B`	0.073 ± 0.007	6, 6, 6
`step70000-tokens147B`	0.122 ± 0.009	6, 6, 6
`step80000-tokens168B`	0.068 ± 0.010	7, 7, 6
`step90000-tokens189B`	0.063 ± 0.005	8, 8, 8
`step100000-tokens210B`	0.073 ± 0.008	7, 7, 6

J Per-Revision DPO: Full Three-Recipe Table

Protocol. We apply three DPO recipes of increasing compute to the same OLMo-2-1B revisions straddling the substrate transition. *Weak*: HH-RLHF 5K pairs, 1 epoch, $\beta = 0.1$ (the §4 recipe) at 5 revisions. *Strong*: HH-RLHF 20K pairs, 2 epochs at 5 revisions. *SFT+DPO*: a two-stage pipeline that SFTs on 20K Tulu-3-SFT-Mixture (Lambert et al., 2024) examples then DPOs on 20K HH-RLHF pairs for 2 epochs, at 5 revisions, a reduced-budget approximation of the shipped OLMo-2-1B-Instruct recipe (omitting the RLVR stage). All three measure $\rho^{(c)}(t)$, $q_{\text{norm}}{}^{(c)}(t)$, selective divergence, and pre-/post-attack refusal per matched pair.

Table 18: Per-revision DPO on OLMo-2-1B under three recipes of increasing strength. $q_{\text{norm}}{}^{\text{ARITH}}$ is the per-recipe layer-max on the ARITH probe. All rows are in the suppression regime by absolute ρ. Weak/Strong show the under-trained-base sub-pattern: pre-substrate $q_{\text{norm}}{}^{\text{ARITH}}$ ($620/863$ at 1B) is more than an order of magnitude above any of their post-substrate values (13 to 48). SFT+DPO shows the SFT-stage-recipe sub-pattern instead: at every post-substrate revision ρ stays elevated (6 to $19\times$ the Weak recipe's) and $q_{\text{norm}}{}^{\text{ARITH}}$ stays high (283 to 770), reflecting the SFT-stage NLL gradient projecting onto F_{cap}'s top-k; at 1B the recipes coincide in ρ. ρ is the mean over blocks; single blocks reach 7.4×10^{-4} (SFT+DPO, 1049B). Table 14 uses a second, independent SFT+DPO run set. Behavioral refusal appears only under SFT+DPO; Weak/Strong produce no measurable refusal at any revision (behavioral columns omitted). This is a weight-space geometry table, not the fragility result: the headline fragility claim is on Qwen/Llama (§4), and the developmental behavioral points that exist show the attack succeeding (SFT+DPO: $21.9\% \to 0\%$ at 1B, $42.2\% \to 15.6\%$ at 4001B; the 105B point, $4.7\% \to 3.1\%$, is within noise), so the table should not be read as "no successful attack." Source for the prose summary in §6.

		Weak		Strong		SFT+DPO			
Step	Tokens	ρ	$q_{\text{norm}}{}^{\text{ARITH}}$	ρ	$q_{\text{norm}}{}^{\text{ARITH}}$	ρ	$q_{\text{norm}}{}^{\text{ARITH}}$	pre-ref	post-ref
3×10^2	1B	8.1×10^{-5}	620	9.5×10^{-5}	863	8.3×10^{-5}	95	21.9%	0%
3×10^4	63B	4.6×10^{-6}	23	6.8×10^{-6}	34	5.8×10^{-5}	603	0%	0%
5×10^4	105B	4.3×10^{-6}	13	6.6×10^{-6}	13	8.0×10^{-5}	283	4.7%	3.1%
5×10^5	1049B	7.2×10^{-6}	33	1.2×10^{-5}	40	1.1×10^{-4}	770	0%	0%
1.9×10^6	4001B	9.8×10^{-6}	22	1.8×10^{-5}	48	5.9×10^{-5}	529	**42.2%**	**15.6%**

K Co-Training: Full Protocol and Per-Condition Tables

Two sets of runs. The four-condition and schedule tables in this appendix and in Table 2 come from the *original* from-scratch runs, trained before the data-loading fix (App. A) and evaluated on 64 AdvBench prompts. The scale runs of Table 3, which the body quotes throughout, are separate

from-scratch runs of `cotrain-continuous` and the LM-only `base`, trained with the corrected loader and scored on 256 prompts. Where both exist the two are different models, not one model scored twice. At 410M `cotrain-continuous` reads $\mathbf{84.4} \rightarrow \mathbf{84.4}\%$ in the original run and $\mathbf{98.4} \rightarrow \mathbf{84.0}\%$ in the scale run: the post-attack values agree and the pre-attack values do not. We take the scale run as the reference: its install erodes modestly rather than not at all, still well short of the 35 to 38 pp that post-hoc installs lose (Table 1). At 1B the two read $87.5 \rightarrow 90.6\%$ (Table 2) and $87.1 \rightarrow 84.8\%$: the pre-attack values agree, and the post-attack gap is about four prompts at $n = 64$, inside that sample's noise (App. A).

Table 19: From-scratch Pythia-410M, 10B C4 tokens, four conditions (referenced from Sec. 7). Refusal rate on 64 AdvBench prompts before and after the benign-SFT attack. This is the original run; the four-condition comparison is valid within it, but for `cotrain-continuous` the separate from-scratch run scored on 256 prompts gives $98.4 \rightarrow 84.0$ rather than the 0 pp shift here, so the erosion is modest rather than absent. Capability is per-probe NLL, where *lower is better*: `cotrain-continuous` improves on the LM-only base on three probes and is within 0.06 nats of it on INSTRUCT. It is the only condition that installs refusal. This run predates the data-loading fix (App. A). Capability of the models behind Table 3 is in Table 20, where ARITH comes out 1.29 nats worse at 410M rather than 1.44 better, a measure of that probe's run-to-run spread.

				capability NLL			
Condition	Pre-ref	Post-ref	Shift	REASON	ARITH	RECALL	INSTRUCT
`base`	0.0%	0.0%	0 pp	5.34	10.01	6.42	5.55
`cotrain-windowed`	0.0%	1.6%	+1.6 pp	4.78	9.11	6.48	5.61
`cotrain-continuous`	**84.4%**	**84.4%**	**0.0** pp	4.95	8.57	6.39	5.60
`post-hoc-dpo`	0.0%	0.0%	0 pp	5.45	10.00	6.42	5.55

Capability at scale (Table 20). Measured on the checkpoints behind Table 3, with the capability probes of §3 kept per example so each co-trained model is paired with its base. Two kinds of difference appear. REASON costs 0.12 to 0.22 nats at every scale from 1B, and RECALL 0.13 to 0.16 at 1B and 2.8B, always in the same direction. ARITH moves by more but changes sign, and its base does not improve monotonically with scale (8.59, 7.97, 8.34, 8.46), so at a 10B-token budget a single run cannot resolve it. INSTRUCT shows no cost anywhere. At an NLL near 4.5 on a one-letter answer the models place about 1% on the correct letter, so REASON here measures the probability of the first answer token more than reasoning; the cost is largest where the answer is shortest. One untested reading is that the refusal templates, which open every safety target, shift probability toward refusal openers at the start of any reply.

Protocol (Pythia-410M from-scratch). Random-initialized Pythia-410M architecture trained on streaming `allenai/c4` for $N \approx 10$B tokens, sequence length 2048, micro-batch 8 per rank, global batch 512 sequences via gradient accumulation across 2 ranks (DDP). AdamW with peak lr 3×10^{-4}, $\beta = (0.9, 0.95)$, weight decay 0.1, gradient clipping 1.0, cosine schedule with 1% warmup. $\mathcal{L}_{\text{safe}}$ is cross-entropy on (AdvBench harmful prompt, refusal-target) pairs: each prompt is paired round-robin with one of 5 short refusal-template targets ("I can't help with that.", "As an AI assistant, I cannot ...", etc.); during safety-active steps each step adds a microbatch of 2 safety pairs weighted by $\lambda(s)$. The training and evaluation prompt pools are both drawn from the public AdvBench CSV under different sampling seeds and partially overlap by construction (training uses 256 prompts from `Random(42)`, evaluation uses 64 from `Random(101)`); this is a deliberate choice, the install we measure is whether the model reproduces *the refusal-template distribution* on harmful prompts, not whether it has memorized prompt-specific responses, so prompt overlap is not a leakage path for that quantity. The two co-training schedules differ only in $\lambda(s)$:

- `cotrain-windowed`: $\lambda(s) = 0.5$ for $s \in [0.10, 0.30]$, $\lambda(s) = 0$ otherwise.
- `cotrain-continuous`: $\lambda(s) = 0.5 \cdot (0.30 - 0.10) = 0.10$ for all s, equal total safety-loss weight to the windowed schedule.

`post-hoc-dpo` runs 5K-pair HH-RLHF DPO on the `base` output. Attack: 200-step / lr 2×10^{-5} benign-SFT on 100 Alpaca examples, evaluated on 64 AdvBench prompts (same recipe as Sec. 4).

Table 20: Capability of the co-trained models at every scale: the eight from-scratch checkpoints behind Table 3, scored on the four capability probes (NLL of the gold answer in nats, natural-log units; lower is better). A cost of c nats means the co-trained model gives the correct answer e^{-c} times the base's probability, equivalently e^{c} times the perplexity: $+0.2$ is about 18% less probability, or 22% higher perplexity. Each co-trained model is compared with its LM-only base on the same examples (126 for REASON, the whole formal_logic test split; 256 for the others), with a 95% paired-bootstrap interval over examples; bold marks an interval that excludes zero. Each condition is one training run, so the interval covers which examples were asked, not run-to-run variance.

	410M	1B	2.8B	6.9B
LM-only base, NLL				
REASON	4.72	4.55	4.52	4.55
ARITH	8.59	7.97	8.34	8.46
RECALL	6.10	5.41	5.07	4.95
INSTRUCT	6.08	5.92	5.82	5.63
co-training cost, continuous − base (nats), with 95% *interval*				
REASON	$+0.02$ $[-0.07, +0.11]$	$\mathbf{+0.20}$ $[+0.12, +0.29]$	$\mathbf{+0.12}$ $[+0.03, +0.22]$	$\mathbf{+0.22}$ $[+0.06, +0.38]$
ARITH	$\mathbf{+1.29}$ $[+1.17, +1.43]$	$+0.05$ $[-0.06, +0.15]$	$\mathbf{+0.67}$ $[+0.55, +0.79]$	$\mathbf{-0.17}$ $[-0.28, -0.05]$
RECALL	$+0.03$ $[-0.05, +0.11]$	$\mathbf{+0.16}$ $[+0.06, +0.24]$	$\mathbf{+0.13}$ $[+0.06, +0.20]$	$+0.01$ $[-0.08, +0.09]$
INSTRUCT	$+0.04$ $[-0.02, +0.11]$	0.00 $[-0.07, +0.06]$	-0.04 $[-0.10, +0.02]$	$+0.03$ $[-0.03, +0.09]$

Protocol (scale runs). The runs of Tables 3, 20 and 22 use the same script and settings as above at every size (10B C4 tokens, sequence length 2048, global batch 512 sequences, peak lr 3×10^{-4} with 1% warmup and cosine decay, AdamW with weight decay 0.1, gradient clipping 1.0, 2 safety pairs per micro-batch), with C4 sharded across data-parallel ranks. 410M and 1B train on 2 GPUs, 2.8B and 6.9B on 4. The learning rate is not retuned per size, and 10B tokens is short of compute-optimal from 1B up (about half the Chinchilla budget at 1B, a fourteenth at 6.9B), so the larger models are under-trained; both arms of each comparison share this.

Geometry across conditions (Table 21). Each non-base condition's geometry is measured against `base` as W_{base}. All three lie in the suppression regime; the (ρ, q_{norm}) instrument does not distinguish behaviorally robust from behaviorally fragile installs at this scale.

Table 21: Geometry of the safety direction at 410M scale, measured against `base`. Mean over 96 (layer, probe) cells. All three conditions sit in the suppression band; behaviorally, only `cotrain-continuous` resists the benign-SFT attack. Refusal is from the original 64-prompt run; the separate scale run of `cotrain-continuous` reads 84.0% at $n = 256$, and the other two conditions have no scale run.

Condition	mean ρ	mean q_{norm}	post-attack refusal
`cotrain-continuous` (uniform)	2.77×10^{-5}	2.31	$\mathbf{84.4}\%$
`cotrain-windowed`	2.75×10^{-5}	2.27	1.6%
`post-hoc-dpo`	3.01×10^{-5}	4.01	0.0%

Where co-trained safety sits relative to capability (Table 22). The Δ-based instrument of Fig. 1a cannot place a co-trained model, for the reason given in Limitation (vii): no post-hoc displacement exists, and the only available Δ is a between-trajectory gap whose small ρ is close to a geometric tautology. The natural substitute reads the safety gradient inside a single model and projects it onto that same model's F_{cap}, giving ρ_{safe} (Table 22). It cannot compare the two arms, because in them it does not measure the same thing. The LM-only `base` has not learned refusal, so its safety gradient is large and points toward installing it. `cotrain-continuous` has already fit the refusal templates, so its safety gradient is 10^4 to 10^5 times smaller, and the direction of a gradient that has nearly vanished says little about where the learned refusal sits. We therefore draw no conclusion from the change in ρ_{safe} between arms, including its rise on INSTRUCT. The between-run numbers of Table 21 show the same limit from the other side: at 410M they put the robust `cotrain-continuous` and the failed `cotrain-windowed` within 2% of each other in

q_{norm} (2.31 against 2.27) while the two differ by 83 pp in refusal, so a geometric predictor of which schedule survives remains open.

Table 22: ρ_{safe} ($\times 10^{-3}$), LM-only `base` → `cotrain-continuous` at the same scale, per capability probe: the safety gradient, taken within one model, projected onto that model's own F_{cap}. Reported for completeness; the two arms are not comparable. The base's safety gradient points toward installing refusal, while the co-trained model, having already fit the refusal templates, has a safety gradient 10^4 to 10^5 times smaller in norm (last column), whose direction carries little information. Absolute values are not comparable to Fig. 1a's ρ or to ρ_{part}, which are calibrated for a displacement rather than a gradient.

Scale	REASON	RECALL	ARITH	INSTRUCT	$\|\nabla\mathcal{L}_{\text{safe}}\|$
410M	$0.93 \to 0.83$	$1.62 \to 1.79$	$0.88 \to 0.96$	$3.56 \to 12.98$	$10.0 \to 3.4\times10^{-4}$
1B	$1.53 \to 1.04$	$2.48 \to 1.98$	$1.86 \to 1.98$	$5.81 \to 7.72$	$4.7 \to 1.1\times10^{-4}$
2.8B	$1.35 \to 0.74$	$1.98 \to 0.43$	$1.75 \to 0.60$	$3.56 \to 23.24$	$2.0 \to 1.5\times10^{-5}$
6.9B	$2.97 \to 2.19$	$3.81 \to 5.20$	$3.79 \to 3.03$	$7.05 \to 14.48$	$1.1 \to 1.0\times10^{-5}$

Stronger attacks (Table 23). The benign-SFT attack of the main text is one point in a family. We sweep its learning rate and step count on the 1B `cotrain-continuous` model, and add the identity-shifting attack of Qi et al. (2024) at the default setting. The worst case over the sweep leaves 75.0% refusal, at the highest learning rate; the identity-shifting attack leaves 100%. Several cells end above the 87.5% pre-attack value, within noise at $n = 64$.

Table 23: Stronger attacks on the 1B `cotrain-continuous` model: post-attack refusal (%, regex) on 64 AdvBench prompts, pre-attack 87.5%. Benign-SFT on 100 Alpaca examples at three learning rates and three step counts; the default attack of the main text is 2×10^{-5} for 200 steps. The identity-shifting attack runs at the default setting.

Learning rate	200 steps	400 steps	800 steps
2×10^{-5}	90.6	90.6	93.8
5×10^{-5}	92.2	85.9	93.8
1×10^{-4}	78.1	**75.0**	84.4
identity-shifting attack		100.0	

160M continued-training pilot. We previously ran a smaller continued-training pilot at Pythia-160M scale (continued-training from the shipped checkpoint on streaming C4 for $N = 2000$ AdamW steps at lr 5×10^{-5}, batch size 4). Table 24 reproduces its four-condition table for completeness; the headline pattern is the same: a 500-step early window (`cotrain-early`) ends with no refusal, while a 500-step late window (`cotrain-late`) installs refusal that the attack *partially erodes*. The from-scratch 410M result above generalizes the same finding: short-window co-training does not produce attack-robust refusal regardless of *when* the window sits during training; only co-training that persists to the end of training does (Table 2).

Table 24: 160M continued-training pilot for completeness. `cotrain-early` ends with no refusal after the LM-only steps that follow its window; `cotrain-late` installs refusal that the attack partially erodes; the two DPO-style conditions install no measurable refusal at 160M (the 5K-pair DPO recipe does not engage). Refusal on 64 AdvBench prompts; this pilot has no 256-prompt rerun and no main-text counterpart.

Condition	Pre-attack refusal	Post-attack refusal	Refusal shift
`baseline`	0.0%	0.0%	0.0 pp
`cotrain-early`	0.0%	1.6%	+1.6 pp
`cotrain-late`	**84.4%**	**15.6%**	**−68.8** pp
`post-hoc-dpo`	0.0%	0.0%	0.0 pp

L Detector Cross-Check: Refusal Gate versus Harmful Output

The main text scores refusal with a regex over refusal-prefix templates. That is the quantity the geometry speaks to: Δ installs a refusal *gate*, and the suppression reading predicts that benign SFT perturbs the gate while leaving the underlying capability routed to the original output distribution. A regex detects whether the gate fires. It does not detect whether what follows is harmful.

Those two questions can come apart, so we rescore every generation behind Tables 1 and 3 with Llama-Guard-3-8B (Inan et al., 2023; Grattafiori et al., 2024), a learned classifier of harmful *output*. Both detectors run on the same saved generations in a single pass, so the columns of Tables 25 and 26 are directly comparable. Table 10 already reports the shipped base models under both.

Table 25: Table 1 under both detectors: refusal (%) before → after the same 100-example benign-SFT attack on the three shipped instruct models. The regex columns are the main-text table. $n = 256/200/256$; the three cells that rise are within noise (App. A).

	AdvBench		HarmBench		StrongReject	
Instruct model	regex	classifier	regex	classifier	regex	classifier
Qwen-2.5-7B	93.0 → 58.2	93.0 → 85.2	67.5 → 45.5	74.0 → 59.5	52.3 → 56.6	72.3 → 73.8
Llama-3-8B	49.2 → 10.9	39.5 → 27.7	46.5 → 6.5	49.0 → 28.0	46.5 → 7.8	53.9 → 32.4
Mistral-7B-v0.3	1.2 → 0.4	12.1 → 6.2	1.5 → 0.0	7.0 → 8.5	3.5 → 0.8	23.0 → 14.1

Table 26: Table 3 under both detectors: from-scratch `cotrain-continuous` refusal (%), pre → post benign-SFT attack, by scale and prompt distribution. The regex columns are the main-text table. $n = 256/200/256$; changes under about 6 pp are within noise (App. A). The lower block is the LM-only base at each scale.

	AdvBench (in-dist.)		HarmBench (out)		StrongReject (out)	
Scale	regex	classifier	regex	classifier	regex	classifier
410M	98.4 → 84.0	91.8 → 64.8	58.0 → 34.5	57.0 → 33.5	8.2 → 7.4	25.8 → 21.5
1B	87.1 → 84.8	83.2 → 63.7	56.0 → 32.0	54.0 → 25.0	7.8 → 8.2	23.0 → 20.3
2.8B	96.1 → 90.2	91.0 → 77.3	45.5 → 31.5	49.0 → 37.5	8.6 → 8.2	31.2 → 19.5
6.9B	**96.9 → 90.6**	**97.3 → 86.7**	32.5 → 39.0	33.0 → 41.5	9.0 → 10.9	29.7 → 19.5
LM-only base						
410M	0.4 → 0.8	6.6 → 5.9	1.5 → 0.5	10.0 → 9.0	0.0 → 1.6	14.8 → 15.6
1B	0.0 → 0.4	9.0 → 7.0	0.0 → 0.0	12.0 → 10.0	0.4 → 1.6	21.5 → 14.8
2.8B	0.4 → 0.0	6.2 → 11.7	0.0 → 0.0	12.5 → 8.5	0.0 → 0.4	24.6 → 20.7
6.9B	0.0 → 0.0	10.9 → 9.8	0.5 → 0.5	13.0 → 10.0	0.0 → 0.4	24.6 → 18.8

What changes, and what does not. The classifier preserves every ordering the main text rests on. Post-hoc installs still lose far more of their refusal to the attack than co-trained ones do, co-training that persists to the end of training is still the only kind that installs refusal (`cotrain-continuous` 82.8% and `cotrain-post-substrate` 85.9%, against 4.7% for `base`, on the 1B schedule runs, Table 27), and the coverage gap between AdvBench and the two out-of-distribution benchmarks still widens with scale.

What the classifier changes is the level, and it does so in both directions. It reads Mistral and Llama as refusing more than the regex does before the attack (Mistral 1.2 against 12.1% on AdvBench), because those models decline in wordings the refusal-prefix regex does not match. It reads the co-trained models as eroding more (`cotrain-continuous` 82.8% to 65.6% on the 1B schedule runs), because after the attack their refusal templates sometimes survive as a prefix while the continuation drifts toward compliance. Both effects are properties of the readout, not of the geometry: the gate and the output are different quantities, and only the first is what a weight-space displacement can be expected to predict. We report the regex in the main text for that reason, and report the classifier here so the gap is visible.

Table 27: The 1B schedule runs of Table 2 under both detectors: refusal (%) before → after the benign-SFT attack, 64 AdvBench prompts. The regex columns are the main-text table's 1B column.

Condition (safety window)	regex	classifier
`cotrain-continuous` $[0, 1]$	$87.5 \to 90.6$	$82.8 \to 65.6$
`cotrain-post-substrate` $[0.30, 1]$	$84.4 \to 84.4$	$85.9 \to 76.6$
`cotrain-windowed` $[0.10, 0.30]$	$3.1 \to 4.7$	$6.2 \to 7.8$
`base` (LM only)	$0.0 \to 0.0$	$4.7 \to 7.8$

M Limitations and Broader Impact

Limitations. (i) F_{cap} is approximated by its top-k eigenspace; tail eigendirections carry some capability mass and are unmeasured, so a weight update with small top-k projection may still traverse low-curvature but semantically important directions that our instrument misses. (ii) Refusal in the main text is scored by a regex over refusal prefixes ("I can't", "as an AI"), which detects whether the refusal gate fires, not whether the output is harmful. App. L rescores every generation with Llama-Guard-3-8B: the classifier shifts levels in both directions, by up to 27 pp (Qwen after the attack), but preserves every ordering the claims rest on, so they rest on relative ranking across models and conditions rather than absolute rates. (iii) The fragility attack is Qi et al.'s benign-fine-tuning variant, the weakest of the three variants in that paper, so our "attack succeeds" claim is conservative. We separately verify that the continuous co-training install is robust to stronger attacks (at 1B, Table 23): a benign-SFT sweep over learning rate and steps leaves worst-case post-attack refusal at 75% and an identity-shifting attack leaves it at 100%; only the harmful-demonstration variant, which needs Qi et al.'s released data, is untested. (iv) The developmental-axis behavioral curve is sparse: SFT+DPO installs substantial pre-attack refusal at only two of five revisions (21.9% at 1B and 42.2% at 4001B, Table 18), too few for a per-revision curve; a denser curve would require a full Tulu-3 pipeline (SFT → DPO → RLVR) at each revision, out of compute budget. (v) The co-training result (Sec. 7) supports continuous-schedule pretraining-time co-training at 410M from-scratch scale, but the head-to-head "robustness at matched pre-attack refusal" comparison against `post-hoc-dpo` is not directly available because the 5K-pair DPO recipe installs no measurable refusal at this scale (0% pre-attack); a stronger post-hoc baseline (SFT+DPO or Tulu-3-style pipeline at 410M) is needed to make the protection differential quantitative. We verify the result from scratch to 6.9B (Table 3); beyond that it remains a compute-bounded conjecture, and the refusal it installs is in-distribution. (vi) The geometric instrument runs to 8B; frontier-scale (70B+) behavior is a conjecture the framework makes but does not verify. (vii) The Δ-based instrument does not apply to the co-training conditions. It characterizes an *edit* $\Delta = W_{\text{safe}} - W_{\text{base}}$ applied to a frozen, converged base: $\|\Delta\| = 0.0015$ for the post-hoc DPO at 410M and 2.9 to 7.8 for the shipped 7 to 8B instruct deltas. A co-trained model was never a capable-but-unsafe base that received a patch, so no such Δ exists; the only available one is the gap between two complete from-scratch runs, $\|\Delta\| = 86.5$. The four conditions share a seed and data order and differ only by the $\lambda\mathcal{L}_{\text{safe}}$ term, so that gap is attributable to the safety loss, but it is four to five orders of magnitude larger than the post-hoc DPO install at the same 410M scale, 11 to 30× larger than the shipped 7 to 8B instruct deltas, and 7 to 11× larger than OLMo's SFT+DPO updates (8.1 to 12.7, Table 14), because adding the term sent the run down a different trajectory: it carries the divergence of the language-modeling path, not a localized safety displacement. Since ρ is scale-free it can still be computed and comes out small ($\approx 3 \times 10^{-5}$), but in a high-dimensional space almost any direction has a small projection onto a k-dimensional subspace, so for a between-trajectory gap that number is close to a geometric tautology rather than evidence of a thin install. This is also why Lemma 1 does not bear on co-training: its premise is a post-hoc displacement from a converged base. Two measurements could settle whether co-trained safety is entangled with capability without differencing two runs. The first reads the safety gradient *within* a single model and projects it onto that model's own F_{cap} (Table 22), but it cannot compare the two arms: a co-trained model has already fit the refusal objective, so its safety gradient is 10^4 to 10^5 times smaller than the LM-only base's, and the direction of a nearly vanished gradient says little about where the learned refusal sits. The second remains open: sweep attack strength and record capability alongside refusal, so that co-trained refusal falling only once capability falls with it would demonstrate an entanglement behaviorally. (viii) Each co-training condition is one training run, so capability differences between conditions are confounded with run-to-run variance. On ARITH two

independent 410M runs disagree by 2.7 nats (Tables 19 and 20); only the REASON and RECALL costs, consistent in sign across scales, are attributable to co-training.

What the instrument does not claim. Beyond the Ethics Statement, one scope limit is worth stating: the instrument flags when an alignment procedure has landed in the suppression regime, a fine-tuning-robustness red flag, not when a model has been made unsafe by any behavioral standard. The two are different questions; only the first is measurable from weight-space geometry.

N Extended Related Work

Post-hoc alignment and its fragility. RLHF and its direct-preference descendants (Christiano et al., 2017; Ouyang et al., 2022; Rafailov et al., 2023) are the standard alignment recipe. Qi et al. (2024) show that only 10 to 100 fine-tuning examples (even benign ones) compromise the safety of aligned GPT-3.5 and Llama-2-chat; the benign-fine-tuning variant is our behavioral attack. Zhan et al. (2024) strip RLHF from GPT-4 with 340 examples and 95% ASR; Andriushchenko et al. (2025) reach 100% ASR on leading aligned models via simple adaptive attacks; Paulus et al. (2025) drive suffix search two orders of magnitude faster. Wei et al. (2023) identify competing-objectives and mismatched-generalization as root causes; Anil et al. (2024) push these to long-context regimes. Closest to our weight-space account: Qi et al. (2025) localize safety to the first few output tokens ("shallow alignment"); Wei et al. (2024) find $\sim$3% of parameters and 2.5% of ranks carry the entire safety install; Peng et al. (2024) map a narrow "safety basin" in weight space around aligned checkpoints. Our contribution is the continuous-geometry account (Fisher-eigenspace regime classification) that these local-pruning / shallow-feature findings are special cases of. Mitigations span prompt-template defenses (Huang et al., 2024a), training-time regularization (Lyu et al., 2024), perturbation-aware alignment (Huang et al., 2024b), representation-level noise defenses (Rosati et al., 2024), and latent adversarial training (Sheshadri et al., 2024); if they too land in the suppression regime, which we do not test, they face the same first-order fragility the Lemma describes.

Mechanistic interpretability of alignment. Arditi et al. (2024) show RLHF-installed refusal is mediated by a single residual-stream direction; Wollschläger et al. (2025) extend this to multi-directional concept cones, and ablation restores full compliance in both cases. Lee et al. (2024) reverse-engineer DPO on toxicity and argue it installs an overlay rather than removing learned features. Zou et al. (2023a); Panickssery et al. (2024) steer behavior via contrastive activation vectors. Templeton et al. (2024); Marks & Tegmark (2024); Lieberum et al. (2024); Marks et al. (2025) find high-level concepts linearly encoded in residual streams of frontier-class models and develop SAE / sparse-circuit tooling to isolate them causally. Hubinger et al. (2024); Greenblatt et al. (2024) provide complementary behavioral evidence that safety training does not erase the underlying capability. Our per-layer ρ and q_{norm} operationalize these activation-level intuitions at the weight level, making the "install vs. remove" distinction geometric and quantitative.

Loss-landscape geometry and Fisher methods. The empirical Fisher matrix F_{cap} is standard in natural-gradient (Amari, 1998; Martens, 2020) and continual-learning literatures, most notably EWC (Kirkpatrick et al., 2017). We use the top-k eigenspace (not the diagonal), which captures multi-parameter correlations that matter when Δ is structured. Grosse et al. (2023) scale influence-function Fisher-Hessian estimates to 52B-parameter models; our top-k Gram trick is a lightweight per-layer variant. Keskar et al. (2017) relate flat minima to generalization; Jacot et al. (2018) give the fixed-kernel infinite-width limit; Ortiz-Jiménez et al. (2023) show task arithmetic becomes exact in the NTK-linearized tangent space, providing the geometric foundation we extend to DPO. Task arithmetic and model merging (Ilharco et al., 2023; Wortsman et al., 2022; Yadav et al., 2023) treat safety- and task-vectors as weight-space objects; Hu et al. (2022) establish the low-intrinsic-rank hypothesis for fine-tuning updates that underlies tractable eigenspace analysis at LLM scale. Our decomposition explains why safety vectors are approximately orthogonal to capability vectors in real-scale models.

Developmental phenomena. Achille et al. (2018) introduce critical learning periods in vision networks; Frankle et al. (2020) localize the most consequential training period to the first few epochs; Kleinman et al. (2024) show the same arises in deep linear networks. Constantinescu et al. (2025) transfer the biological critical-period analogy directly to language models and find

catastrophic-forgetting windows concentrated in the first $\sim 10\%$ of continued pretraining; our 1B to 63B substrate transition in OLMo-2-1B sits in the same developmental bracket. LLM emergent abilities (Wei et al., 2022; Srivastava et al., 2023; Schaeffer et al., 2023) turn on sharply at scale- or data-dependent thresholds, and the quantization-model view (Michaud et al., 2024) describes them as discrete "quanta" acquired during training; Chinchilla-style scaling (Hoffmann et al., 2022) places our substrate transition (1B to 63B tokens) around the compute-optimal budget for a 1B model, about 20B tokens. We connect these to safety: not only do capabilities emerge at a specific pretraining time, but the substrate that lets *safety* bind to those capabilities has its own developmental transition.

Safety signal during pretraining. Korbak et al. (2023) inject preference feedback into pretraining objectives; Bai et al. (2022b) train from constitutional principles; Zhou et al. (2023) argue alignment needs only a small high-quality SFT. Our co-training result (Sec. 7) is complementary: a windowed-vs-continuous comparison at matched total safety-loss weight shows that *persistence* of the safety signal across pretraining, not its timing, drives attack robustness.

Unlearning as contrast. Machine-unlearning methods attempt to genuinely remove capability rather than install a refusal gate. Li et al. (2024) introduce the WMDP benchmark and the RMU method, which randomizes representations for targeted harmful knowledge; Lynch et al. (2024) benchmark eight unlearning methods and find capability is often reliably re-extractable. The contrast sharpens our framing: unlearning targets the erasure regime ($\rho \gtrsim \rho_{\text{part}}$), DPO/RLHF target the suppression regime; neither in isolation solves the fragility problem, consistent with Lemma 1 (erasure requires motion *inside* the capability subspace, which suppression-regime DPO lacks).

Open-weight checkpoint suites. Pythia (Biderman et al., 2023) pioneered intermediate-checkpoint releases; OLMo-2 (Team OLMo et al., 2024) ships 267 checkpoints plus a Tulu-3 (Lambert et al., 2024)-trained Instruct variant with real refusal behavior (the combination we need for the developmental axis). Cross-family replications use Qwen-2.5 (Yang et al., 2024), Llama-3 (Grattafiori et al., 2024), and Mistral-7B-v0.3 (Jiang et al., 2023).