SFT LEAVES A REPRODUCIBLE SIGNATURE IN ACTIVATION GEOMETRY: CALIBRATING POST-TRAINING INTERPRETABILITY AGAINST RUN-TO-RUN NOISE

Anonymous authors
Paper under double-blind review

ABSTRACT

A model diff credits post-training with whatever separates a base checkpoint from one post-trained descendant, presuming a rerun would land close by. On the sparse co-activation graph of `Qwen3-1.7B` it does not: two LoRA-SFT reruns sit over half as far apart as either sits from the base. We score each base-to-post contrast against its own recipe's run-to-run band, and show that under run exchangeability k runs certify no better than $p \leq 1/(k+1)$ however large the effect, so the band's z is a descriptive scale. SFT leaves a large, reproducible signature: base→SFT *displaces* the graph by 19.9 pairwise-distance SDs beyond the mean distance between four independent LoRA-SFT runs, and the pre-registered estimator replicates on Llama-3.1-8B with Llama-Scope. At an identical 29 steps, four SFT and four DPO runs occupy *disjoint* displacement ranges, a gap $5.3\times$ either arm's spread, without any band. Three other readings stop short of a verdict: DPO's band membership is dose-dependent ($0/4$, $2/4$, $4/4$ contrasts outside at $12, 29, 60$ steps) while its displacement stays flat; light GRPO is a secondary non-detection, not an absence, and we run no equivalence test; and the graph concentration read as an RL signature is already in the zero-RL base, and a dictionary-width control removes an apparent scaling law in it. A pre-registered causal sweep (three edge budgets, $4{,}816$ held-out prompts) found the displaced sources load-bearing ($12\times$ a matched random control in mean, $3.5\times$ in median at $10{,}000$ edges), but their stage-specificity interaction reversed at two budgets and the third failed its faithfulness gate, so we claim calibrated structural displacement, not a changed mechanism. Model diffs should be read beside their recipe's rerun band, within-band diffs as non-detections. **Scope:** the full detection protocol covers one backbone (`Qwen3-1.7B`), one SFT recipe and general-text probe states; the matched-base control replicates across three families.

1 INTRODUCTION

Model diffing reads what post-training did off one base checkpoint and one post-trained descendant, such as Gemma-2-2B and its chat model (Minder et al., 2025) or Qwen2.5-Math-7B and its reasoning-distilled variant (Hu et al., 2026); each is one run of its recipe, so the reading presumes a second run would land close to it. Diff the sparse co-activation graph of `Qwen3-1.7B` before and after 29 steps of LoRA-SFT and the graph moves: across five SFT runs (the audited one and four reseeded reruns), its $10{,}000$ strongest edges shift by an edge-Jaccard distance of 0.059–0.063. With only the seed changed, two SFT runs already sit 0.034–0.038 apart, more than half the base-to-SFT distance. LoRA-SFT's diff exceeds every rerun distance by a wide margin, on two backbones; DPO's band verdict changes with step count while its displacement does not; and light GRPO is not resolved from its reruns.

Interpretability work reads post-training off exactly this kind of diff: a statistic on the post-trained checkpoint, sometimes differenced against its base, is reported as what SFT, preference optimization or RL did to the model (Zhang et al., 2025a; Hu et al., 2026; Shi et al., 2026), often against the expectation that post-training reweights what the base already computes (Yue et al., 2025; Wang

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.

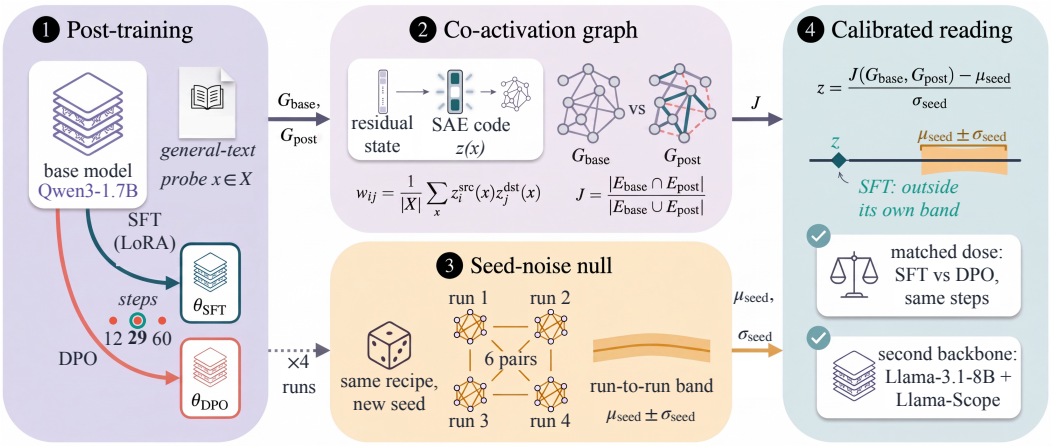


Figure 1: **A post-training change counts only if it exceeds what re-running the same recipe produces.** (1) The base model (Qwen3-1.7B) is post-trained by LoRA-SFT or DPO; every checkpoint is probed on the same general-text set. (2) Last-position states are encoded by the base-trained SAE, edges are weighted by mean co-activation, and base and post graphs are compared by edge Jaccard J. (3) Four independent same-recipe runs give six pairwise distances, the run-to-run band. (4) Each contrast is scored against its own recipe's band, $z = (J - \mu_{\text{seed}})/\sigma_{\text{seed}}$, a descriptive scale rather than a tail probability; the diamond stands for SFT, more displaced than every seed pair (our 'outside' rule); $\mu_{\text{seed}} \pm \sigma_{\text{seed}}$ only sets the z scale. DPO's band membership depends on dose ($0/4$, $2/4$, $4/4$ contrasts outside at $12/29/60$ steps) and is not summarized by the marker. Schematic only; measured values in Fig. 2.

et al., 2025; Lin et al., 2024) while policy entropy collapses (Cui et al., 2025). The feature-level "narrowing" prediction we audit is our own extrapolation of that picture to sparse features; those papers make behavioral claims. Two comparisons are usually missing: the same statistic on the base, and the spread between runs of the same recipe. Without them, base properties and run-to-run variation are reported as effects of training; with both in place, an RL signature in graph concentration turns out to be a base property and a scaling law in it a dictionary-width artifact.

We build the missing comparisons. A *matched-base control* credits post-training only with the change in a statistic from the base, computed with the same SAE and probes. A *seed-noise null* scores each base-to-post displacement against the spread of independently seeded runs of the recipe that produced it, with the same estimator, so both are measured alike (Fig. 1). A *positive control* (the instrument must detect some stage, so that its non-detections mean something), a random-dictionary control and a precision rule complete the audit (Secs. 3–4.2). The seeded LoRA runs are re-seeded before model construction, so their bands include adapter initialization and not data order alone; the full-parameter GRPO runs share their starting weights.

What survives, and what does not. SFT's signature survives: LoRA-SFT displaces the co-activation graph far outside its own four-run band ($z = -19.9$, a descriptive scale) and again on Llama-3.1-8B with Llama-Scope, and against the GRPO-family band it is detected under a random dictionary as under the trained SAE. At the identical 29 optimizer steps, four SFT and four DPO runs occupy disjoint displacement ranges, a comparison that needs no band. Three readings do not survive as verdicts: DPO's band membership depends on dose while its displacement stays flat (the band moves, the effect does not grow; Table 1), so we report the curve; the concentration that single-checkpoint studies read as an RL signature is already far above a permutation null in the zero-RL base, with the GRPO graph at the same level; and light GRPO is not resolved from its run-noise band, a non-detection and not an equivalence result, although it did learn its task (accuracy gain on 100 held-out items, Sec. 4.2). We draw no behavioral/internal dissociation from the cross-state comparison. Nor do we read displacement as mechanism: a registered causal sweep found the stage-specificity interaction reversed at two of three budgets.

Contributions.

- **A rerun-band protocol for model diffing**: each base-to-post displacement is scored against same-recipe reruns with the same estimator, single-checkpoint statistics are credited only with their change from the base, and a precision rule fixes what can be scored (Sec. 3); the supplement recomputes the headline statistics without a GPU. The permutation rank floor applies: k runs certify no better than $p \leq 1/(k+1)$ under run exchangeability, whatever the number of seed pairs (Proposition 1), so z is a descriptive scale.
- **A reproducible SFT signature**: base→SFT displaces the co-activation graph beyond its own run-to-run band on two backbones and two SAE families, is disjoint from DPO at an equal step count without any band, and its dropped edges concentrate on boilerplate-to-hub edges (Sec. 4.2).
- **Four ways a signature reading fails, each with the control that catches it**: base-set concentration, a TopK-pinned sparsity metric, a width-driven "scaling law", and a DPO band that moves with dose while displacement stays flat, so its verdict depends on the dose (Secs. 4.1–4.2).

2 Related Work

What post-training changes inside a model. The comparison our band calibrates, one base against one post-trained run, is how this literature measures fine-tuning. Mechanistic studies mostly find reuse of circuitry the pretrained model already has. Jain et al. (2024) describe fine-tuning on procedurally defined tasks as learning a thin wrapper over existing capabilities, Prakash et al. (2024) find entity tracking improved by strengthening the same circuit, and Lee et al. (2024) show DPO lowers toxicity by bypassing rather than removing the responsible directions. For RL-era post-training the picture is less settled. Shi et al. (2026) train SFT and RL from one base and find RL's feature changes restrained next to SFT's; Rojas Nunez et al. (2026) find SFT disrupts head-level circuits more than Dr.GRPO (Liu et al., 2025); Zhang et al. (2025a) report RL raising activation intensity and diversity, with DPO departing from that trend; Sinha et al. (2026) compare six alignment algorithms across three families. Our question comes before theirs: how large must such a difference be before it exceeds what a second run of the same recipe produces?

Model diffing with sparse dictionaries. Sparse autoencoders (Bricken et al., 2023; Templeton et al., 2024; Gao et al., 2025), transcoders (Dunefsky et al., 2024) and attribution graphs (Ameisen et al., 2025; Lindsey et al., 2025; Marks et al., 2025) give a basis in which two checkpoints can be compared feature by feature. Diffing work builds on them: crosscoders (Lindsey et al., 2024) separate base-only from chat-only latents once their sparsity artifacts are corrected (Minder et al., 2025), stage-wise fine-tuning of a base SAE isolates the features a fine-tune changes (Bricken et al., 2024), dedicated-feature crosscoders localize RL-induced tool use (Shportko et al., 2026), and transcoder adapters diff a base model against its reasoning-distilled variant (Hu et al., 2026). Closest to our SFT result, Minder et al. (2026) show that narrow fine-tuning leaves biases in activation differences, read on the first tokens of random text, strong enough to recover the fine-tuning domain, and that mixing pretraining data into the fine-tuning set largely removes them. Our SFT set is narrow in their sense (900 function-calling rows), so its signature may be such a bias; what we add is the rerun band its size is judged against, and the same test applied to DPO and GRPO. We use the plainest member of the family, a base-trained SAE applied to every checkpoint with a co-activation graph over its codes, because a calibration result should not depend on a dictionary fitted to the models being compared; a random-dictionary control (Korznikov et al., 2026) checks that it does not depend on the trained dictionary either. Following Munigety (2026), we treat attribution mass as correlational and do not read displacement as a changed mechanism.

Run-to-run variability as the reference. Representations are not reproducible across seeds by default. Networks trained from different initializations learn partly different units (Li et al., 2016); BERT reproductions that differ only in seed support different conclusions (Sellam et al., 2022); fine-tuning outcomes vary with initialization and data order alone (Dodge et al., 2020). The instruments vary too: SAEs trained on identical data share roughly 30% of their features (Paulo & Belrose, 2025), individual features are seed-unstable while their span is not (Gerasimov et al., 2026), apparent dictionary stability can be an initialization artifact (Brzozowski & Chung, 2026), and SAE benchmark scores carry more reseed noise than assumed (Chanin, 2026). Méloux et al. (2025) argue that interpretability findings should be tested against explicit alternatives because the standard tools

explain even random networks. These works measure seed variation in outcomes, units or dictionaries. We use the seed variation of the post-training run itself, read through the diffing estimator, as the reference a base-to-post difference must exceed.

Behavioral and weight-space accounts of RL post-training. Pass@k analyses argue that RLVR sharpens sampling within the base model's support rather than expanding it (Yue et al., 2025); a single training example recovers much of the RLVR gain (Wang et al., 2025); base models align in context without weight updates (Lin et al., 2024); and RLHF trades output diversity for generalization relative to SFT (Kirk et al., 2024). Weight- and state-space accounts point the same way: RLVR weight deltas are close to rank one (Wei et al., 2026), post-training can be read as reshaping the state distribution (Nie, 2026), and base models already hold the reasoning mechanisms that thinking models learn to deploy (Venhoff et al., 2026). None of these measures sparse features, and none says how large an internal footprint RL should leave. The seed-calibrated audit supplies that reference for one regime, light GRPO on a 1.7B model.

3 Problem Setup

Runs. A recipe A (objective, data, hyperparameters, step count) maps the base model M_0 and a seed s to a checkpoint M_s^A. The base is `Qwen3-1.7B` (Yang et al., 2025), post-trained on BFCL function calling (Patil et al., 2025): four LoRA-SFT runs (rank-16, α=32 on the attention projections, 0.37% of parameters, 29 steps), four LoRA-DPO runs (Rafailov et al., 2023) at 12 steps plus four retrained runs read at 29 and 60 steps, and six full-parameter GRPO-family runs (Shao et al., 2024) (an accuracy-only baseline, four further seeds, one shaped-reward variant, 100–300 steps, so that band mixes seed, step and reward variance; L7). Every run starts from M_0; the seeded LoRA runs are re-seeded before model construction, so they differ in adapter initialization and not only in data order. A second backbone (`Llama-3.1-8B` (Grattafiori et al., 2024), four LoRA-SFT runs) and the public Tulu-3 staircase complete the roster (App. D).

Co-activation graph. Let X be a fixed probe set (n=500 Pile prompts (Gao et al., 2020), last-position residual states) and $f^{\text{src}}, f^{\text{dst}}$ SAE encoders trained on M_0 only (Bricken et al., 2023; Templeton et al., 2024; Gao et al., 2025) (`qwen-scope` (Deng et al., 2026), TopK with k=50, d=32,768, layers 14 $\to$ 22). With codes $z^{\text{src}}(x; M) = f^{\text{src}}(h_{14}(x; M)) \in \mathbb{R}^d_{\geq 0}$ and likewise z^{dst},

$$w_{ij}(M) = \frac{1}{|X|} \sum_{x \in X} z_i^{\text{src}}(x; M)\, z_j^{\text{dst}}(x; M)\, \mathbf{1}[(i,j) \in T_x], \quad E(M) = \text{top}_N\, w(M). \tag{1}$$

Here $E(M)$ keeps the $N = 10{,}000$ largest weights, and T_x holds the 2,000 largest products at state x (of at most 2,500 non-zero ones), as in the builder in the supplement. Because one basis serves every checkpoint, $E(M)$ and $E(M')$ are subsets of the same index set. Base-trained SAEs usually transfer to chat-tuned checkpoints (Kissane et al., 2024); here the base SAE reconstructs base, SFT and GRPO states equally well (App. E). The graph is a co-activation proxy, not a gradient or causal attribution; App. J shows an edge can move while the causal coupling it summarizes does not.

Levels and the matched-base control. A single-graph statistic T is either the Gini coefficient of inbound weights $W_j = \sum_i w_{ij}$ over active destination features, or the share of $\sum_j W_j$ held by the top 1% of them. Single-checkpoint studies read $T(M_s^A)$; we attribute to post-training only $\Delta T = T(M_s^A) - T(M_0)$, computed with the same SAE and probes. A degree-preserving permutation of the weights answers a different question (concentrated versus chance).

Displacement and the own-run band. Edge Jaccard is $J(M, M') = |E(M) \cap E(M')| / |E(M) \cup E(M')|$ and displacement is $D = 1 - J$. For k runs of recipe A, the contrasts are $c_s = J(M_0, M_s)$ and the seed pairs are $d_{st} = J(M_s, M_t)$ for $1 \leq s < t \leq k$, $m = k(k-1)/2$ of them. We summarize the band by

$$\mu_{\text{seed}} = \frac{1}{m} \sum_{s<t} d_{st}, \qquad \sigma^2_{\text{seed}} = \frac{1}{m-1} \sum_{s<t} \big(d_{st} - \mu_{\text{seed}}\big)^2, \qquad z_s = \frac{c_s - \mu_{\text{seed}}}{\sigma_{\text{seed}}}, \tag{2}$$

and call c_s *outside* the band when $c_s < \min_{s<t} d_{st}$, reporting $K_{\text{out}} = \#\{s : c_s < \min d\}$ out of k. The m pairs share runs, so z_s is a distance scale, not a tail probability. *Precision rule:* a statistic is

scored only if σ_{seed} exceeds its reporting step $\delta = 0.0001$. We therefore commit to **edge Jaccard** as the scored statistic ($\sigma_{\text{seed}} > 11\delta$ on every band in Table 1); the weighted edge cosine fails the rule on trained dictionaries and is reported only descriptively (App. E).

Dose. For DPO every quantity in (2) is indexed by the step count $\tau \in \{12, 29, 60\}$. The mean displacement $\bar{D}(\tau) = 1 - \frac{1}{4}\sum_s c_s(\tau)$ is flat across the three doses, while $\mu_{\text{seed}}(\tau)$ rises (seed-pair D falls) and $\sigma_{\text{seed}}(\tau)$ falls, then holds (Table 1, Fig. 2b; values in App. E). Hence $K_{\text{out}}(\tau) = 0, 2, 4$ of 4 although the contrasts $c_s(\tau)$ are flat: band membership is dose-dependent because the seed-pair band rises (enlarging $\mu_{\text{seed}} - c_s$), not because the contrasts move further from the base.

What a band can certify. Model the null "the stage changes the graph no more than re-running the recipe does" as exchangeability: under it the base behaves as one more run.

Proposition 1 (Band exceedance under run exchangeability)**.** *Let $R_0, \ldots, R_k$ $(k \geq 2)$ be random checkpoints whose joint law is invariant under permutations of the indices, let S be any symmetric similarity, and let $\mathcal{P} = \{\{s,t\} : 1 \leq s < t \leq k\}$. Whatever the dependence among the $m = k(k-1)/2$ pairs,* (a) $\Pr\left[S(R_0, R_1) < \min_{\{s,t\}\in\mathcal{P}} S(R_s, R_t)\right] \leq 1/k$*; and* (b) $\Pr\left[\max_{1\leq s\leq k} S(R_0, R_s) < \min_{\{s,t\}\in\mathcal{P}} S(R_s, R_t)\right] \leq 1/(k+1)$*. Both bounds are attained by some exchangeable law.* (c) *If n_1 runs of one recipe and n_2 of another are jointly exchangeable, their displacements from a fixed base occupy disjoint ranges with probability at most $2\, n_1!\, n_2!/(n_1+n_2)!$.*

The intuition: the band is built from only k runs, and any of the $k+1$ runs could have been the odd one out. Part (b) is the rank floor of permutation tests and conformal p-values (Phipson & Smyth, 2010; Angelopoulos & Bates, 2023); what the proposition adds is that the m seed pairs, though more numerous than the runs, do not lower it, and that both bounds are attained. So the strongest guarantee that band exceedance gives under exchangeability alone is $p \leq 1/k$ for one contrast and $p \leq 1/(k+1)$ when all k are outside, however large $|z|$ is (proof in App. A). Three design choices follow. With $k = 4$, SFT's four seeded runs, all outside, give $p \leq 1/5$ under this null and no smaller (the audited checkpoint, seeded differently, is not one of the k), so we quote z descriptively and rest the SFT claim on its size, its replication on a second backbone and SAE family, and the band-free comparison of part (c). A single public run pair ($k = 2$) gives at best $p \leq 1/3$, which is why the Tulu-3 staircase is a consistency check (Sec. 4.3). And a rank quantile $1/(m+1)$, which treats the contrast as exchangeable with the m seed pairs, does not follow from run exchangeability; we report such ranks as ranks. The null is a modelling choice: every run starts from M_0, so the base is the runs' common origin rather than an exchangeable draw, and we do not claim that the bound is conservative for edge Jaccard on these checkpoints. A contrast at band level also cannot rule out a change all runs share, so we read within-band contrasts only as non-detections.

4 Results

4.1 Single-checkpoint levels are set by the base model

The Qwen base graph is highly concentrated: the top 1% of active destination features receive 42.7% of inbound weight, far above a degree-preserving permutation null. A single-checkpoint reading of the GRPO graph would call that an RL signature. The matched base removes it (Fig. 3): the zero-RL base has the same concentration and nearly all of the same top hubs, and every light Qwen recipe stays within a narrow Gini band of its base with near-identical hubs (App. H). `Gemma-2-2B` (Gemma Team et al., 2024) with `gemma-scope` (Lieberum et al., 2024) shows the same base-set level; its heavily instruction-tuned `-it` checkpoint moves further (Fig. 3b), on a two-point contrast possibly confounded by the chat template (App. F).

The activation-side narrowing battery is base-dominated too. The same control applies to the metrics read as narrowing: firing-frequency Gini, mean-activation Gini and usage entropy (Table 8). Mean L_0 is uninformative by construction on `qwen-scope`: a TopK dictionary reads its k on every checkpoint, so we drop it where sparsity is fixed and report it only for JumpReLU dictionaries (Rajamanoharan et al., 2024). On the metrics that can move, light GRPO is base-dominated (relative changes under 1%, on Pile and on GSM8K (Cobbe et al., 2021) probes), and heavy instruction tuning on Gemma moves its two largest-moving metrics against narrowing (L_0 rises and firing-frequency

Table 1: **SFT moves the graph further than its own reruns on two backbones; DPO's band reading depends on dose; light GRPO is not resolved from a wider band.** Each arm has k independently seeded runs and $k(k-1)/2$ seed pairs; entries are displacements $D = 1 - J$ (Eq. 2). "Outside" counts contrasts more displaced than every seed pair; z is descriptive (Proposition 1). Receipts: `notes/circuit_rewiring_*.json` (App. E).

Backbone	Arm	k	seed-pair D	σ_{seed}	contrast D	outside	z
Qwen3-1.7B	LoRA-SFT, 29 steps	4	0.0344–0.0381	0.0014	0.0588–0.0630	$5/5^a$	-20.0 to -17.0
Llama-3.1-8B	LoRA-SFT	4	0.0222–0.0257	0.0013	0.0434–0.0444	$4/4$	—[b]
Qwen3-1.7B	LoRA-DPO, 12 steps	4	0.0294–0.0357	0.0022	0.0327–0.0354	$0/4$	-1.11 to $+0.11$
	LoRA-DPO, 29 steps	4	0.0303–0.0334	0.0012	0.0323–0.0365	$2/4$	-3.61 to -0.01
	LoRA-DPO, 60 steps	4	0.0294–0.0329	0.0012	0.0340–0.0359	$4/4$	-4.10 to -2.56
Qwen3-1.7B	GRPO family[c]	6	0.0336–0.0394	0.0016	0.0361–0.0406	$1/6$	-2.0 to $+0.8$

Matched dose, no band: SFT 0.0588–0.0630 vs. DPO 0.0323–0.0365 at 29 steps; gap $0.0223 = 5.3\times$ either arm's spread (0.0042).

[a] four seed runs plus the audited checkpoint. [b] the registered rule is band exceedance; no z is quoted. [c] baseline, four further seeds, one shaped reward, 100–300 steps (L7).

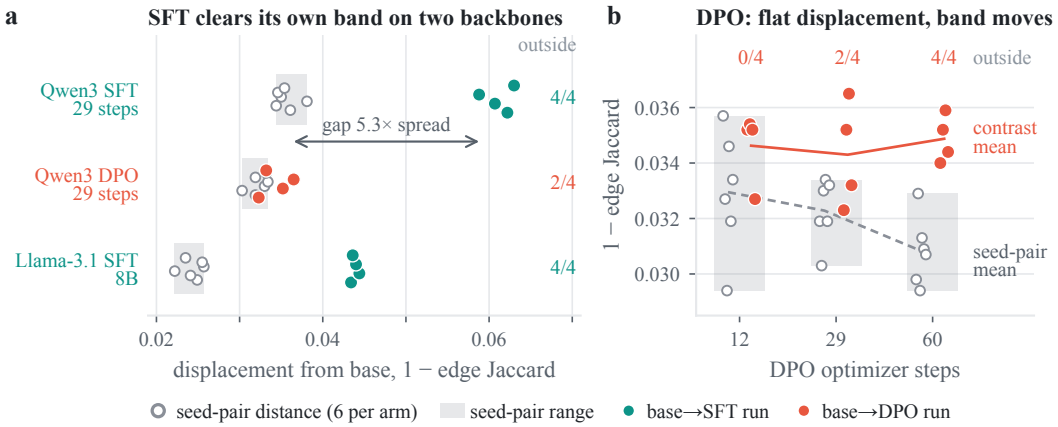


Figure 2: **The opening question answered: SFT moves the graph further than its reruns do, while DPO's verdict turns on the band, not the displacement.** (a) Open circles are the six seed-pair distances of each arm and shading spans their range (not a confidence interval); filled points are the four seeded base-to-post runs of each arm (Table 1 also counts the audited SFT checkpoint, hence its $5/5$). At the same 29 steps the SFT and DPO runs occupy disjoint ranges; the arrow is the gap between them, $5.3\times$ either arm's own range (0.0223 vs. 0.0042). (b) DPO read at three doses, as separate arms because the registered trajectory-continuity check failed (App. E), so the lines join arm means, not one trajectory: the contrast mean (solid) is flat while the seed-pair mean (dashed; both in D) falls (its spread narrows from 12 to 29 steps, then holds), so $0/4$, $2/4$, $4/4$ contrasts end up outside. Receipts as in Table 1; generator `code/analysis/fig_main_seednull_v2.py`.

Gini falls, while the other two barely move; Table 8). A dictionary-width denominator had manufactured a second artifact, an apparent scaling law in concentration, that width-independent statistics remove (App. H).

4.2 The seed-noise-null audit

SFT is detected against its own runs. Scored against its six seed pairs, base→SFT sits at $z = -19.9$; all five SFT contrasts fall outside the band (Table 1, Fig. 2a). Scoring SFT against the GRPO band would have been the borrowed-null substitution this audit exists to catch; the SFT runs resemble each other slightly more, and more consistently, than GRPO runs do, so the right null strengthens the SFT reading rather than weakening it (App. E). Three checks keep this from being an artifact of the instrument. A random-direction encoder at matched width and sparsity reproduces

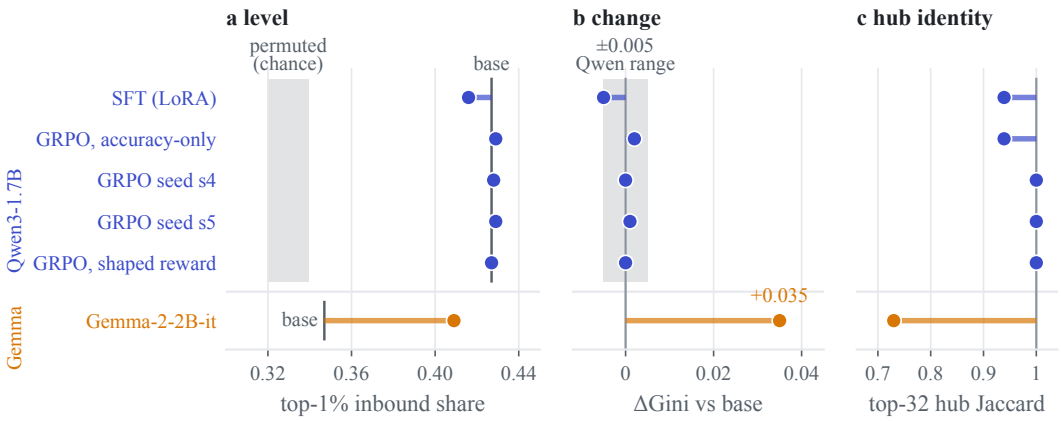


Figure 3: **The concentration a single-checkpoint study would credit to RL is already present in the zero-RL base.** One row per checkpoint in the base-diff summary. (a) Top-1% inbound share: every Qwen checkpoint sits on or next to its base (vertical rule), far from the degree-preserving permutation null of the base graph (95% range shaded). (b) Change in Gini against the base; the shading is $\pm$ the largest $|\Delta\text{Gini}|$ among the five Qwen rows (0.005), not a statistical null. (c) Top-32 hub Jaccard against the base. Gemma-2-2B-it is a different family and a heavier recipe; the comparison is descriptive. Receipts: `notes/basediff_{qwen,gemma}_summary.json`, `h1_permutation_null_1p7b_l14to22.json`.

it against the GRPO-family band (Table 3; one random draw), so the detection does not need the trained dictionary; only the feature naming does. The band construction does not depend on the statistic, so crosscoder latents or activation-difference readouts can be scored against it the same way (we have not run them); we score the co-activation graph because the two readings we retract were made on it. Subsets of 2,000 and 5,000 edges keep SFT detected. The pre-registered Llama-3.1-8B replication puts all four contrasts beyond its band, the nearest at 1.7$\times$ the band maximum. LoRA touches a small fraction of parameters while the GRPO runs are full-parameter, so parameter count does not explain the ordering, and on the Tulu-3 staircase the full-parameter SFT stage writes the largest increment (Table 2, no band); objective, data and duration still differ between the stages.

Where the displacement lands. Of the edges that the base and all six GRPO runs keep but SFT drops, most, carrying 63% of the dropped weight, feed a single destination: the base graph's largest hub, a generic numeric, whitespace and markup feature that fires on most tokens. The sources losing the most weight are boilerplate and function-token features, while the edges SFT adds are diffuse, spread more evenly over their destinations than the drops (App. J). A plausible reading is instruction tuning learning to down-weight boilerplate. A causal probe keeps that reading correlational: ablating the source features moves the hub up to several times more than count-matched random features (an upper bound; the control is not magnitude-matched), but the effect is suppressive and no weaker in SFT than in base, so the edge drop is a change in co-activation statistics rather than a pruned causal coupling (App. J).

The SFT/DPO ordering needs no band. At SFT's 29 steps the gap between the four SFT and four DPO runs is several times either arm's spread (Table 1), and SFT's mean displacement is well above DPO's (App. E). Disjoint ranges are the most extreme of the 70 equal splits of eight runs, so under exchangeable runs the two-sided permutation probability is $1/35$ (Proposition 1c; computed after the fact, not registered). Four matched-construction DPO runs with per-run data shuffling land inside the fixed-order 29-step contrast range, with none outside their own band, and stay disjoint from SFT (App. E). The arms still differ in learning rate and dropout, so this compares two standard recipes at matched adapter geometry and step count, not the objectives in isolation (L8).

DPO's band reading depends on dose. At the registered 12 steps DPO passes its positive control (its displacement is several times the SFT band's 95% width; App. E) and every contrast stays inside its own band (the registered no-separation branch). The pre-registered ladder then puts half and then

Table 2: **At production scale SFT and DPO write most of the displacement, but one checkpoint per stage gives no seed band to score it against** (Tulu-3 staircase, Llama-3.1-8B, Llama-Scope layers 16→25, n=500, unnormalized v1 operating point; at the normalized one ΔGini $\leq$ 0.019, App. L). Left: levels vs. base (base Gini 0.780). Right: stage increments; the reference is one PPO/GRPO run pair (0.903), not a seed distribution. Receipts: `notes/tulu_*.json` (v1; `tulu_v2_*` normalized).

Stage (vs. base)	ΔGini	hub Jacc	Stage increment	edge Jacc	hub Jacc
SFT	+0.009	0.73	base→SFT	0.669	0.73
DPO	+0.004	0.60	SFT→DPO	0.796	0.73
RLVR-PPO	+0.005	0.60	DPO→PPO	0.958	0.94
RLVR-GRPO	+0.005	0.56	DPO→GRPO	0.906	0.88

all of the contrasts outside at the two longer doses with displacement flat (Fig. 2b). A band that moves while the effect stands still will eventually separate anything, and the DPO band lacks the data-order variance the SFT band contains (the shuffled runs above, which include it, stay inside their band; the 60-step arm was not rerun this way), so we follow the branch registered for doses that disagree: report the curve and promote no single z. Training is not bitwise reproducible at fixed seed, so the registered continuity check failed and the doses are separate arms; each band is a genuine run-to-run null (App. E).

Light GRPO is not resolved from its run noise. Against the 15 pairs of six GRPO-family runs, base→GRPO and the other RL contrasts sit inside the band except one marginal case (s1; Table 1), which falls just outside it, by a small fraction of SFT's margin, and at matched step 300 sits just outside the three-pair same-step band (inside the 15-pair one); within one run, displacement does not grow from step 100 to 300 (App. E). This band mixes seeds, steps and a reward variant, which is conservative for detection and anti-conservative for certifying absence (L7); with no equivalence margin defined, these are non-detections; at $|z| \geq 2$ this band flags an edge-Jaccard shift of 0.0032, about one-eighth of SFT's shift against it (App. E). GRPO did learn its task: BFCL accuracy rises over base, with a 95% interval that excludes zero (n=100; App. E). Accuracy is measured on task states and displacement on general text, and the matched on-task instrument fails its own positive control, so the pair supports no behavioral/internal comparison (L5, App. I).

Displacement is not mechanism. Structurally distinct circuits can compute the same function (Bayat Makou et al., 2026), so we registered their causal test and ran it at three edge budgets on 4,816 held-out prompts. Corrupting the sources unique to the SFT graph moves SFT's output far more than frequency-matched random sources (12× in mean, 3.5× in median at 10,000 edges), but relative to the base-specific sources they matter *less* in SFT than in base (the interaction J is negative at 5,000 and 10,000 edges, with CIs excluding zero; Table 12), and at 2,000 the base-specific sources fail the gate (registered verdict: MIXED). We retire the word *rewiring* (App. K). Two auxiliary probes read each checkpoint's *generated text* through a fixed external reader (Pythia-70m (Biderman et al., 2023) with a public SAE (Marks et al., 2025)) rather than the audited activations; activation deltas stay polarity-preserving and the per-feature entropy change stays below the collapse gate we set in advance (App. M).

4.3 Production scale: the Tulu-3 staircase

A public pipeline releases one checkpoint per stage and algorithm, so it shows what a diff can say without a seed band. On the public `Llama-3.1-8B` → Tulu-3 SFT → DPO → RLVR pipeline (Lambert et al., 2025; Rafailov et al., 2023; Schulman et al., 2017), read through base-trained `Llama-Scope` SAEs (He et al., 2024), levels again stay close to the base (Table 2), and here L_0 is a real measurement (JumpReLU) that barely moves (App. F). Cumulative displacement is large, with top-32 hub overlap falling along the staircase. SFT and DPO write most of it. The RLVR increments sit at or above the similarity of two public RLVR runs branched from the same DPO parent, a reference that also differs in algorithm and recipe. It is a single run pair with no dispersion, and by Proposition 1 it could certify at best $p \leq 1/3$, so the staircase is a production-scale consistency check, not a calibrated null.

5 DISCUSSION

The opening diff can mislead in a small number of recurring ways, and each has a control. A statistic can be large, significant and scale-stable and still be a property of the base model (Sec. 4.1). A denominator tied to dictionary width can produce a scaling law from nothing. A null whose SD sits below the reporting step turns rounding into significance, which is why the precision rule exists. And agreement between groups on a direction (Shi et al., 2026; Rojas Nunez et al., 2026) supplies no width; only same-recipe runs do. Proposition 1 adds a quantitative version of the last point: what a band can certify grows with the number of runs, not with the effect size, so for a fixed budget more seeds buy more than a larger z. In numbers: under run exchangeability, one recipe's band certifies $p \leq 0.05$ only with $k \geq 19$ reruns, all outside (part b), whereas two recipes with four runs each reach $p \leq 1/35$ when their displacement ranges are disjoint (part c), a cheaper certificate for a different question (recipe versus recipe, not base versus post). Where a band verdict changes with dose, as DPO's does, contrast and band should be reported as curves over steps rather than as one verdict. Our nulls do not resolve a footprint of light GRPO at 1.7B from its run-to-run variation, and this non-detection neither measures that footprint nor confirms the small-footprint weight-space and state-distribution accounts above (Wei et al., 2026; Nie, 2026): GRPO's contrasts sit at seed-pair distance (Table 1), and a within-band contrast cannot exclude a change that all runs share (Sec. 3).

What the audit does not show matters as much. Displacement beyond the band says the co-activation structure changed; it does not say the computation changed, and our registered causal sweep found the stage-specificity interaction reversed at two of three budgets. The base-set hubs admit a pretraining-crystallized reading, consistent with persona directions that form early in pretraining and still steer the post-trained model (Moskvoretskii et al., 2026), but we have tested it only through the matched base. And the GRPO non-detection covers light, accuracy-only GRPO at 1.7B read on general text; long-horizon math RLVR at scale is outside this envelope.

6 LIMITATIONS

Nine scope conditions bound the claims (full statements in App. F). **(L1)** the full detection protocol covers one backbone (1.7B), one SFT recipe and general-text probe states; the one cross-family post checkpoint for the light-recipe level control (`gemma-2-2b-it`) is heavy instruct-tuning. **(L2)** the top-32 hub set is chosen by the inbound weight whose concentration it summarizes, not at random. **(L3)** the two generated-text probes re-tokenize each checkpoint's output for the Pythia reader, and that step has its own noise floor. **(L4)** the top-1% statistic is not width-comparable. **(L5)** the on-task (BFCL) arm is *uninformative*: its 2σ flag threshold is $\sim 3\times$ SFT's pile effect ($\sim 4\times$ in relative terms) and the positive control does not separate; hence no dissociation claim. **(L6)** the OLMo-2 (Team OLMo et al., 2025) fourth staircase is blocked: no public base-trained SAE weights. **(L7)** the GRPO band conflates seed, step and one reward recipe — wider than pure seed noise: conservative for detection, *anti*-conservative for the RL null. **(L8)** the DPO evidence is one LoRA recipe at 1.7B, with lr $10\times$ below SFT's and dropout 0.0 vs. SFT's 0.05; its band membership is dose-dependent, so neither a single z nor a dose-independent detection or non-detection is promoted, and the public Tulu-3 staircase has only one DPO checkpoint, read against the RLVR run-pair reference (App. E). **(L9)** the behavioral check is n=100; the SFT gain is not resolved from zero on it.

7 CONCLUSION

Against same-recipe reruns, LoRA-SFT's change is large and reproducible: it displaces the co-activation graph by 19.9 descriptive pairwise-distance SDs beyond its own four runs' mean, exceeds its band again on a second backbone and SAE family, and at the identical step count its four-run range is *disjoint* from DPO's without a band. LoRA-DPO's displacement stays flat over three doses while its band membership changes, and light GRPO is not resolved from its run noise. A pre-registered causal sweep finds the displaced sources load-bearing but their stage-specificity interaction reversed at two of three budgets, so calibration buys structural change, not mechanism. Model diffs should take single-checkpoint statistics as changes from the base, sit beside their recipe's rerun band (as curves where the verdict moves with dose), treat within-band diffs as non-detections, and spend budget on reruns: under run exchangeability, k runs certify no better than $p \leq 1/(k+1)$.

REFERENCES

Emmanuel Ameisen, Jack Lindsey, Adam Pearce, Wes Gurnee, Nick L. Turner, Brian Chen, Craig Citro, David Abrahams, Shan Carter, Basil Hosmer, et al. Circuit tracing: Revealing computational graphs in language models. *Anthropic Transformer Circuits Thread*, 2025. URL https://transformer-circuits.pub/2025/attribution-graphs/methods.html.

Anastasios N. Angelopoulos and Stephen Bates. Conformal prediction: A gentle introduction. *Foundations and Trends in Machine Learning*, 16(4):494–591, 2023. doi: 10.1561/2200000101.

Alireza Bayat Makou, Jingcheng Niu, Subhabrata Dutta, and Iryna Gurevych. Many circuits, one mechanism: Input variation and evaluation granularity in circuit discovery. *Transactions on Machine Learning Research*, 2026. arXiv:2606.06267.

Stella Biderman, Hailey Schoelkopf, Quentin Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning (ICML)*, 2023. arXiv:2304.01373.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, et al. Towards monosemanticity: Decomposing language models with dictionary learning. Anthropic Transformer Circuits Thread, 2023. URL https://transformer-circuits.pub/2023/monosemantic-features.

Trenton Bricken, Siddharth Mishra-Sharma, Jonathan Marcus, Adam Jermyn, Christopher Olah, Kelley Rivoire, and Thomas Henighan. Stage-wise model diffing. *Transformer Circuits Thread*, 2024. URL https://transformer-circuits.pub/2024/model-diffing/index.html.

Michał Brzozowski and Neo Christopher Chung. Ablating archetypes: The stability of archetypal SAEs is an artifact of initialization and metric design. *arXiv preprint arXiv:2606.02061*, 2026.

Jorge A. Castillo, Marco Torres Yévenes, and Juan Carlos Lanas. From direction to magnitude: How multimodal instruction-tuning reorganizes the geometric encoding of identity-specifying prompts in transformer hidden states. *arXiv preprint arXiv:2607.09842*, 2026.

David Chanin. Are sparse autoencoder benchmarks reliable? arXiv preprint arXiv:2605.18229, 2026.

Runjin Chen, Andy Arditi, Henry Sleight, Owain Evans, and Jack Lindsey. Persona vectors: Monitoring and controlling character traits in language models. *arXiv preprint arXiv:2507.21509*, 2025.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Ganqu Cui, Yuchen Zhang, Jiacheng Chen, Lifan Yuan, Zhi Wang, Yuxin Zuo, Haozhan Li, Yuchen Fan, Huayu Chen, Weize Chen, Zhiyuan Liu, Hao Peng, Lei Bai, Wanli Ouyang, Yu Cheng, Bowen Zhou, and Ning Ding. The entropy mechanism of reinforcement learning for reasoning language models. *arXiv preprint arXiv:2505.22617*, 2025.

Boyi Deng, Xu Wang, Yaoning Wang, Yu Wan, Yubo Ma, Baosong Yang, Haoran Wei, Jialong Tang, Huan Lin, Ruize Gao, Tianhao Li, Qian Cao, Xuancheng Ren, Xiaodong Deng, An Yang, Fei Huang, Dayiheng Liu, and Jingren Zhou. Qwen-Scope: Turning sparse features into development tools for large language models, 2026. arXiv:2605.11887.

Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Hannaneh Hajishirzi, and Noah Smith. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping. *arXiv preprint arXiv:2002.06305*, 2020.

Jacob Dunefsky, Philippe Chlenski, and Neel Nanda. Transcoders find interpretable LLM feature circuits. In *NeurIPS*, 2024. arXiv:2406.11944.

Asvin G. and Jack Lindsey. From simulation to enaction: Post-trained language models recognize and react to their own generations, 2026. arXiv:2605.25459.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The Pile: An 800GB dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2406.04093.

Gemma Team, Morgane Riviere, Shreya Pathak, et al. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

Gleb Gerasimov, Timofei Rusalev, Nikita Balagansky, Daniil Laptev, Vadim Kurochkin, and Daniil Gavrilov. Unstable features, reproducible subspaces: Understanding seed dependence in sparse autoencoders, 2026. arXiv:2606.12138.

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

Michael Hanna, Mateusz Piotrowski, Jack Lindsey, and Emmanuel Ameisen. circuit-tracer. https://github.com/decoderesearch/circuit-tracer, 2025.

Zhengfu He, Wentao Shu, Xuyang Ge, Lingjie Chen, Junxuan Wang, Yunhua Zhou, Frances Liu, Qipeng Guo, Xuanjing Huang, Zuxuan Wu, et al. Llama Scope: Extracting millions of features from Llama-3.1-8B with sparse autoencoders. *arXiv preprint arXiv:2410.20526*, 2024.

Stefan Heimersheim and Neel Nanda. How to use and interpret activation patching. *arXiv preprint arXiv:2404.15255*, 2024.

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 *International Conference on Learning Representations (ICLR)*, 2022. arXiv:2106.09685.

Nathan Hu, Jake Ward, Thomas Icard, and Christopher Potts. Transcoder adapters for reasoning-model diffing, 2026. arXiv:2602.20904.

Kexin Huang, Haoming Meng, Junkang Wu, Jinda Lu, Chiyu Ma, Ziqian Chen, Xue Wang, Bolin Ding, Jiancan Wu, Xiang Wang, Xiangnan He, Guoyin Wang, and Jingren Zhou. On the direction of RLVR updates for LLM reasoning: Identification and exploitation, 2026. arXiv:2603.22117.

Samyak Jain, Robert Kirk, Ekdeep Singh Lubana, Robert P. Dick, Hidenori Tanaka, Edward Grefenstette, Tim Rocktäschel, and David Scott Krueger. Mechanistically analyzing the effects of fine-tuning on procedurally defined tasks. In *ICLR*, 2024.

Nick Jiang, Isaac Kauvar, and Jack Lindsey. The value axis: Language models encode whether they're on the right track, 2026. arXiv:2606.17056.

Prasanth K K. Feature geometry of LoRA adapters: A sparse autoencoder analysis of representational divergence in fine-tuned language models. *arXiv preprint arXiv:2605.28896*, 2026.

Robert Kirk, Ishita Mediratta, Christoforos Nalmpantis, Jelena Luketina, Eric Hambro, Edward Grefenstette, and Roberta Raileanu. Understanding the effects of RLHF on LLM generalisation and diversity. In *ICLR*, 2024.

Connor Kissane, Robert Krzyzanowski, Arthur Conmy, and Neel Nanda. SAEs (usually) transfer between base and chat models. AI Alignment Forum, https://www.alignmentforum.org/posts/fmwk6qxrpW8d4jvbd/saes-usually-transfer-between-base-and-chat-models, 2024.

Anton Korznikov, Andrey Galichin, Alexey Dontsov, Oleg Rogov, Ivan Oseledets, and Elena Tutubalina. Sanity checks for sparse autoencoders: Do SAEs beat random baselines? *arXiv preprint arXiv:2602.14111*, 2026.

Akshay Krishnamurthy, Audrey Huang, and Nived Rajaraman. Select and improve: Understanding the mechanics of post-training for reasoning, 2026. arXiv:2606.13125.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V. Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, Yuling Gu, Saumya Malik, Victoria Graf, Jena D. Hwang, Jiangjiang Yang, Ronan Le Bras, Oyvind Tafjord, Chris Wilhelm, Luca Soldaini, Noah A. Smith, Yizhong Wang, Pradeep Dasigi, and Hannaneh Hajishirzi. Tulu 3: Pushing frontiers in open language model post-training. In *Conference on Language Modeling (COLM)*, 2025. arXiv:2411.15124.

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. In *ICML*, 2024.

Gengyang Li, Zheng-Fan Wu, Siqi Bao, and Yunfang Wu. Not all tokens learn alike: Attention entropy reveals heterogeneous signals in RL reasoning, 2026. arXiv:2605.07660.

Xiaomin Li, Zhou Yu, Zhiwei Zhang, Xupeng Chen, Ziji Zhang, Yingying Zhuang, Narayanan Sadagopan, and Anurag Beniwal. When thinking fails: The pitfalls of reasoning for instruction-following in LLMs. arXiv preprint arXiv:2505.11423, 2025.

Yixuan Li, Jason Yosinski, Jeff Clune, Hod Lipson, and John Hopcroft. Convergent learning: Do different neural networks learn the same representations? In *ICLR*, 2016.

Tom Lieberum, Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Nicolas Sonnerat, Vikrant Varma, János Kramár, 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.

Bill Yuchen Lin, Abhilasha Ravichander, Ximing Lu, Nouha Dziri, Melanie Sclar, Khyathi Chandu, Chandra Bhagavatula, and Yejin Choi. The unlocking spell on base LLMs: Rethinking alignment via in-context learning. In *ICLR*, 2024. arXiv:2312.01552.

Jack Lindsey, Adly Templeton, Jonathan Marcus, Thomas Conerly, Joshua Batson, and Christopher Olah. Sparse crosscoders for cross-layer features and model diffing. *Transformer Circuits Thread*, 2024. URL https://transformer-circuits.pub/2024/crosscoders/index.html.

Jack Lindsey, Wes Gurnee, Emmanuel Ameisen, Brian Chen, Adam Pearce, Nick Turner, Craig Citro, David Abrahams, Shan Carter, Basil Hosmer, et al. On the biology of a large language model. *Anthropic Transformer Circuits Thread*, 2025. URL https://transformer-circuits.pub/2025/attribution-graphs/biology.html.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding R1-Zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*, 2025.

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.

Maxime Méloux, Giada Dirupo, François Portet, and Maxime Peyrard. The dead salmons of AI interpretability, 2025. arXiv:2512.18792.

Julian Minder, Clément Dumas, Caden Juang, Bilal Chughtai, and Neel Nanda. Overcoming sparsity artifacts in crosscoders to interpret chat-tuning. In *NeurIPS*, 2025. arXiv:2504.02922.

Julian Minder, Clément Dumas, Stewart Slocum, Helena Casademunt, Cameron Holmes, Robert West, and Neel Nanda. Narrow finetuning leaves clearly readable traces in activation differences. In *International Conference on Learning Representations (ICLR)*, 2026. arXiv:2510.13900.

Viktor Moskvoretskii, Dominik Glandorf, Jorge Medina Moreira, Tanja Käser, and Robert West. Tracing persona vectors through LLM pretraining, 2026. arXiv:2605.13329.

Caleb Munigety. From correlation to cause: A five-stage methodology for feature analysis in transformer language models, 2026. arXiv:2605.22462.

Mahdi Nasermoghadasi. Reading task failure off the activations: A sparse-feature audit of GPT-2 small on indirect object identification. *arXiv preprint arXiv:2605.22719*, 2026.

Dong Nie. Post-training is about states, not tokens: A state distribution view of SFT, RL, and on-policy distillation, 2026. arXiv:2605.22731.

Shishir G. Patil, Huanzhi Mao, Fanjia Yan, Charlie Cheng-Jie Ji, Vishnu Suresh, Ion Stoica, and Joseph E. Gonzalez. The Berkeley function calling leaderboard (BFCL): From tool use to agentic evaluation of large language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 48371–48392, 2025.

Gonçalo Paulo and Nora Belrose. Sparse autoencoders trained on the same data learn different features. *arXiv preprint arXiv:2501.16615*, 2025.

Belinda Phipson and Gordon K. Smyth. Permutation P-values should never be zero: Calculating exact P-values when permutations are randomly drawn. *Statistical Applications in Genetics and Molecular Biology*, 9(1), 2010. doi: 10.2202/1544-6115.1585.

Nikhil Prakash, Tamar Rott Shaham, Tal Haklay, Yonatan Belinkov, and David Bau. Fine-tuning enhances existing mechanisms: A case study on entity tracking. In *ICLR*, 2024.

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 *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. arXiv:2305.18290.

Senthooran Rajamanoharan, Tom Lieberum, Nicolas Sonnerat, Arthur Conmy, Vikrant Varma, János Kramár, and Neel Nanda. Jumping ahead: Improving reconstruction fidelity with JumpReLU sparse autoencoders. *arXiv preprint arXiv:2407.14435*, 2024.

Jeanmely Rojas Nunez, Viraj Sawant, Nathan Allen, Nomgondalai Amgalanbaatar, Yannis Zongo, Vasu Sharma, and Maheep Chaudhary. Mechanistic origins of catastrophic forgetting: why RL preserves circuits better than SFT? arXiv preprint arXiv:2605.28860, 2026.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Thibault Sellam, Steve Yadlowsky, Jason Wei, Naomi Saphra, Alexander D'Amour, Tal Linzen, Jasmijn Bastings, Iulia Turc, Jacob Eisenstein, Dipanjan Das, Ian Tenney, and Ellie Pavlick. The MultiBERTs: BERT reproductions for robustness analysis. In *ICLR*, 2022.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, et al. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. verl: Volcano engine reinforcement learning for LLMs. https://github.com/volcengine/verl, 2024.

Dan Shi, Zhuowen Han, Simon Ostermann, Renren Jin, Josef van Genabith, and Deyi Xiong. Why does reinforcement learning generalize? A feature-level mechanistic study of post-training in large language models. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, 2026.

Andrii Shportko, Shubham Bhokare, Ahmed Zeyad A. Alzahrani, Bowen Cheng, Gustavo Mercier, and Jessica Hullman. Localizing RL-induced tool use to a single crosscoder feature, 2026. arXiv:2606.26474.

Reece Shuttleworth, Jacob Andreas, Antonio Torralba, and Pratyusha Sharma. LoRA vs full fine-tuning: An illusion of equivalence. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. arXiv:2410.21228.

Aarush Sinha, Ishan Garg, Veeraraju Elluru, Arth Singh, and Kushal Garg. Mechanistic analysis of alignment algorithms in language models. *arXiv preprint arXiv:2606.09850*, 2026.

Wendy K. Tam. The neutral mask: How alignment training provides shallow alignment while leaving partisan structure intact in a large language model, 2026. arXiv:2606.09735.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, Nathan Lambert, et al. 2 OLMo 2 furious. In *Conference on Language Modeling (COLM)*, 2025. arXiv:2501.00656.

Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, et al. Scaling monosemanticity: Extracting interpretable features from Claude 3 Sonnet. *Anthropic Transformer Circuits Thread*, 2024. URL https://transformer-circuits.pub/2024/scaling-monosemanticity/.

Constantin Venhoff, Iván Arcuschin, Philip Torr, Arthur Conmy, and Neel Nanda. Base models know how to reason, thinking models learn when. In *International Conference on Machine Learning (ICML)*, 2026. arXiv:2510.07364.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart Shieber. Investigating gender bias in language models using causal mediation analysis. *Advances in Neural Information Processing Systems*, 33:12388–12401, 2020.

Tianle Wang, Jiayu Liu, Zhongyuan Wu, Shenghao Jin, Wei Chen, Hao Xu, and Ning Miao. Linear dynamics in the RLVR training of large language models, 2026a. arXiv:2601.04537.

Xuekang Wang, Zhuoyuan Hao, Shuo Hou, Hao Peng, Juanzi Li, and Xiaozhi Wang. Reproducing, analyzing, and detecting reward hacking in rubric-based reinforcement learning, 2026b. arXiv:2606.04923.

Yiping Wang, Qing Yang, Zhiyuan Zeng, Liliang Ren, Liyuan Liu, Baolin Peng, Hao Cheng, Xuehai He, Kuan Wang, Jianfeng Gao, Weizhu Chen, Shuohang Wang, Simon Shaolei Du, and Yelong Shen. Reinforcement learning for reasoning in large language models with one training example. In *NeurIPS*, 2025. arXiv:2504.20571.

Zhepei Wei, Xinyu Zhu, Wei-Lin Chen, Chengsong Huang, Jiaxin Huang, and Yu Meng. You only need minimal RLVR training: Extrapolating LLMs via rank-1 trajectories, 2026. arXiv:2605.21468.

Xuansheng Wu, Xiaoman Pan, Wenlin Yao, and Jianshu Chen. Soundness-aware level: A microscopic signature that predicts LLM reasoning potential, 2025. arXiv:2510.15216.

Lecheng Yan, Ruizhe Li, Guanhua Chen, Qing Li, Jiahui Geng, Wenxi Li, Longyue Wang, and Chenyang Lyu. Spurious rewards paradox: Mechanistically understanding how RLVR activates memorization shortcuts in LLMs. In *International Conference on Machine Learning (ICML)*, 2026. arXiv:2601.11061.

An Yang, Anfeng Li, Baosong Yang, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Junyao Yang, Chen Qian, Kun Wang, Linfeng Zhang, Quanshi Zhang, Yong Liu, and Dongrui Liu. Entropy-gradient inversion: Moving toward internal mechanism of large reasoning models, 2026. arXiv:2605.17770.

Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Yang Yue, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in LLMs beyond the base model? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. arXiv:2504.13837.

Eslam Zaher, Maciej Trzaskowski, Quan Nguyen, and Fred Roosta. The geometric wall: Manifold structure predicts layerwise sparse autoencoder scaling laws, 2026. arXiv:2605.09887.

Honglin Zhang, Qianyue Hao, Fengli Xu, and Yong Li. Reinforcement learning fine-tuning enhances activation intensity and diversity in the internal circuitry of llms, 2025a. arXiv:2509.21044.

Yunxiang Zhang, Muhammad Khalifa, Lechen Zhang, Xin Liu, Ayoung Lee, Xinliang Frederick Zhang, Farima Fatahi Bayat, and Lu Wang. Logit arithmetic elicits long reasoning capabilities without training, 2025b. arXiv:2507.12759.

Zekai Zhao, Qi Liu, Kun Zhou, Zihan Liu, Yifei Shao, Zhiting Hu, and Biwei Huang. Activation control for efficiently eliciting long chain-of-thought ability of language models, 2025. arXiv:2505.17697.

Ethics Statement

This work uses public language models and public corpora, involves no human subjects and no personal data, and makes no deployment claim. Post-training diffs are routinely read as evidence about what an alignment procedure did; the ethical risk we address is that such readings are made without a noise floor, so that ordinary run-to-run variation is reported as a property of the method. We therefore calibrate every claim against repeated runs of the same algorithm and state which effects exceed that floor and which do not. Any use of this signature to audit a model's training history inherits the scope limits stated in the limitations section.

Reproducibility Statement

Protocols, the run roster and hyperparameters are in App. D. The anonymous supplementary material contains every co-activation graph and receipt JSON named in the paper; the readers and checkers that recompute the headline SFT, DPO, matched-construction and Llama statistics without a GPU (its `README.md` lists the commands); the graph builder; the figure generators; the SFT and DPO trainers with their small BFCL-derived inputs, job specifications and the DPO-loss unit tests; and all pre-registrations. It does not contain model checkpoints, the GRPO training pipeline or the software environments, so it supports recomputation of the reported statistics, not bit-for-bit end-to-end retraining (App. D).

Use of AI Assistants

We used generative AI tools (Anthropic Claude and OpenAI Codex agents) to implement experiment code and check it for correctness, to collect public datasets, to check references, and to polish the writing. The authors take full responsibility for all content.

A Proof of Proposition 1

Write $S(a,b)$ for $S(R_a, R_b)$ and $[k]_0 = \{0, 1, \ldots, k\}$. Say a pair $\{u, v\}$ *avoids* a if $a \notin \{u, v\}$. No assumption is made about the dependence among the similarities: pairs sharing a run may be arbitrarily correlated.

Part (a). For an ordered pair (a, b) of distinct indices in $[k]_0$ define the event

$$E_{ab} = \big\{ S(a,b) < S(u,v) \text{ for every pair } \{u,v\} \subseteq [k]_0 \text{ that avoids } a \big\}.$$

The pairs avoiding 0 are exactly the seed pairs $\mathcal{P}$, so E_{01} is the event in (a). For any permutation π of $[k]_0$ with $\pi(0) = a$, $\pi(1) = b$, E_{ab} is E_{01} evaluated on $(R_{\pi(0)}, \ldots, R_{\pi(k)})$, which has the law of $(R_0, \ldots, R_k)$; hence $\Pr(E_{ab}) = \Pr(E_{01})$ for all $k(k+1)$ ordered pairs. Let $N = \sum_{(a,b)} \mathbf{1}[E_{ab}]$. Then $k(k+1)\Pr(E_{01}) = \mathbb{E}N$, and (a) follows from the deterministic claim $N \le k + 1$.

Step 1: if E_{ab} and E_{cd} both hold with $a \neq c$, then $d = a$ or $b = c$. Otherwise $\{c, d\}$ avoids a, so E_{ab} gives $S(a,b) < S(c,d)$, and $\{a, b\}$ avoids c, so E_{cd} gives $S(c,d) < S(a,b)$, a contradiction.

Step 2: true events have at most two distinct first indices. Suppose E_{ab}, E_{cd}, E_{gh} hold with a, c, g distinct. If $b = c$, Step 1 on (a, g) forces $h = a$ (since $b = c \neq g$), and then on (c, g) forces $d = g$; the events are E_{ac}, E_{cg}, E_{ga}, and since each of the pairs $\{c, g\}, \{g, a\}, \{a, c\}$ avoids the first index of the preceding event, $S(a, c) < S(c, g) < S(g, a) < S(a, c)$, a contradiction. If $b \neq c$, Step 1 on (a, c) forces $d = a$, then on (c, g) forces $h = c$, then on (a, g) forces $b = g$; the events are E_{ag}, E_{ca}, E_{gc} and the same argument gives the cycle $S(a, g) < S(g, c) < S(c, a) < S(a, g)$.

Step 3: counting. With one first index a there are at most k true events (one per $b \neq a$). With two, a and c, let B and D be the sets of second indices. By Step 1, every $b \in B$ and $d \in D$ satisfy $d = a$ or $b = c$. If some $b \in B$ differs from c, then $D \subseteq \{a\}$ and $N \leq |B| + 1 \leq k + 1$; otherwise $B \subseteq \{c\}$ and $N \leq 1 + |D| \leq k + 1$.

Part (b). Define $F_a = \{S(a, x) < S(u, v)$ for every $x \neq a$ and every pair $\{u, v\}$ avoiding $a\}$; F_0 is the event in (b), and by exchangeability all F_a have the same probability. For $a \neq c$, pick $g \notin \{a, c\}$ (possible since $k + 1 \geq 3$): F_a gives $S(a, g) < S(c, g)$ and F_c gives $S(c, g) < S(a, g)$. So the $k + 1$ events are disjoint and $\Pr(F_0) \leq 1/(k + 1)$.

Attainment. Let U be uniform on $[k]_0$ and $\xi_0, \ldots, \xi_k$ i.i.d. uniform on $(0, 1)$, independent of U, and set $R_i = (\mathbf{1}[i = U], \xi_i)$, which is exchangeable. Let $S(R_i, R_j) = \xi_j$ if $i = U$, ξ_i if $j = U$, and $1 + \xi_i \xi_j$ otherwise. Then F_0 holds exactly when $U = 0$, so (b) holds with equality. For (a), E_{01} holds when $U = 0$, and when $U = 1$ it holds exactly when $\xi_0 < \xi_j$ for all $2 \leq j \leq k$ (probability $1/k$); for $U \notin \{0, 1\}$ it fails because the pairs $\{U, j\}$ avoid 0 and have similarity below 1. So $\Pr(E_{01}) = \frac{1}{k+1} + \frac{1}{k+1} \cdot \frac{1}{k} = \frac{1}{k}$.

Part (c). With the base fixed, the displacements $D_i = 1 - S(M_0, R_i)$ of $n = n_1 + n_2$ jointly exchangeable runs are exchangeable. For each of the $n!/(n_1!\, n_2!)$ subsets L of size n_1, the event "every D_i, $i \in L$, strictly exceeds every D_j, $j \notin L$" has the same probability, and these events are disjoint; the same holds with the inequality reversed. The observed labelling is one subset, so disjoint ranges in either order have probability at most $2\, n_1!\, n_2!/n!$, with equality for continuous i.i.d. displacements. For $n_1 = n_2 = 4$ this is $2/70 = 1/35$.

Scope. Exchangeability of the base with the runs is the formal content of "no change beyond run-to-run variation"; it is a modelling choice, not something the base checkpoint satisfies literally. The proposition bounds what a band can certify under that null. It says nothing about power, and it does not turn a descriptive z into a tail probability.

B Additional related-work comparisons

B.1 Independent convergence on the SFT-over-RL ordering

Independent convergence on the direction. Rojas Nunez et al. (2026) introduce *differential circuit vulnerability*, a head-level measure of how much a circuit degrades under fine-tuning, and report on `Qwen2.5-3B-Instruct` that SFT adapts faster to the target task but produces substantially greater circuit disruption, while RL (Dr.GRPO, applied on top of their SFT model) preserves a larger fraction of the base circuit. That is the ordering our dose-matched arm finds, reached independently through a different backbone, a different mechanistic unit (attention heads, not SAE features), and a different comparator (no DPO). We read the *direction* — SFT writes more than the compared alternative (Dr.GRPO there, DPO here; neither group tested the other's comparator) — as converging across groups. What is not settled is how much of the ordering survives calibration against run-to-run variance — the question this protocol answers, and why we report the SFT–DPO separation as disjoint four-run ranges (App. G).

B.2 The null-calibration gap

The null-calibration gap in the RL-internals literature. The gap has teeth: Nasermoghadasi (2026) report a Holm-corrected $|d|$=2.93 SAE finding whose top feature survives seed replication in 1 of 5 runs; Paulo & Belrose (2025) measure $\sim$30% cross-seed feature overlap on identical data. Sinha et al. (2026) span six preference-optimization methods and three model families, using probes,

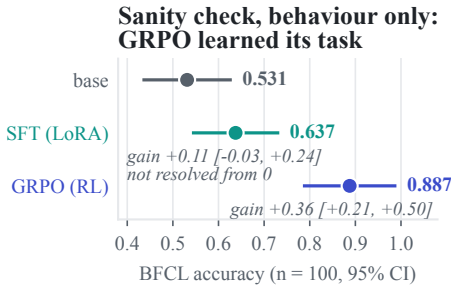


Figure 4: **Sanity check: GRPO did learn the behavior it was trained for.** On the held-out BFCL validation split (100 items, 40/20/20/20 over four categories, from the same release files as the training rows; accuracy is the unweighted category mean), GRPO improves accuracy $+0.36$ over base (95% CI $[+0.21, +0.50]$); SFT's $+0.11$ is not resolved from zero by this harness ($[-0.03, +0.24]$). Intervals are stratified normal approximations and the gains are unpaired (`notes/bfcl_behavior_ci.json`, `code/scripts/bfcl_behavior_ci.py`). *Only behavior is plotted — there is deliberately no internal axis here*, because placing the two on one pair of axes is the withdrawn dissociation claim drawn: accuracy is measured on BFCL task states, displacement on general text, and the matched on-task internal arm fails its own positive control (App. I, L5). The corresponding internal readings are reported where they are calibrated (Sec. 4.2), not here.

SAEs and crosscoders, yet do not compare between-algorithm differences with the spread across repeated post-training runs; Castillo et al. (2026) calibrate a fingerprint by permutation but with one model per regime, confounding regime with family. A 2025–26 wave of mechanistic work argues RL/RLVR changes model internals, each with real but partial controls — and, to our knowledge, *none* calibrates its claimed change against the run-to-run variability of the training algorithm itself. Shi et al. (2026) is the closest prior art (RL induces *restrained* feature changes vs. SFT's many new features) but does not ask whether the residual changes exceed seed noise: every detection in this wave is a *difference read at a single run* (App. G). Zhang et al. (2025a) report the sharpest apparent conflict, and also report that DPO deviates from their trends; we do not read that as confirming our dose-dependent DPO reading (App. G).

C Behavioral sanity-check visualization

Figure 4 plots the behavioral sanity check; provenance and intervals are in App. E.

D Setup and reproducibility

Compute. Single-node L40S/A100/H100 (48–80 GB), total budget approximately 35 GPU-hours across all v1 results (a notebook estimate from job wall-clocks, with no committed accounting receipt and no itemization, so whether it covers the original GRPO training runs is not recorded; each co-activation graph or metric-battery pass takes < 10 minutes on an L40S).

Backbones, checkpoints, and SAEs (v1 protocol). Base: `Qwen3-1.7B`. Post-training checkpoints trained by us (the SFT, DPO and step-300 s1 runs for this study; the other GRPO-family runs earlier, on the same task; task: BFCL function calling from the gorilla repository's release files `BFCL_v4_*`, single-turn categories `simple_python`, `multiple`, `parallel` and `parallel_multiple`, split per category (seed 0) into the training rows and a held-out validation split; accuracy outcome-correctness reward; the SFT/DPO training scripts are in the anonymous supplement, `code/training/` and `code/scripts/slurm/`): SFT (LoRA r=16, α=32 on `q,k,v,o_proj` — 6,422,528 trainable parameters, 0.37% of the model; 900 BFCL SFT rows, 1 epoch = 29 optimizer steps, lr 5e-5, batch 8×accum 4, bf16; merged at analysis; the recipe is the one in `code/training/sft_warmup.py` and `code/scripts/slurm/run_sig_sft_seed_null.sbatch` — no `adapter_config.json` is committed), plus *four further seeded LoRA-SFT*

runs (seeds 0–3, identical recipe and step count, built for the SFT seed null of Sec. 4.2; `code/scripts/slurm/run_sig_sft_seed_null.sbatch`, graphs `notes/basediff_sft_seed{0..3}_l14to22.json`. Seeding note: the audited checkpoint set only the trainer seed, which fixes data order but not the LoRA initialization, so the four null runs are re-seeded before model construction via `code/scripts/sft_seed.py`; seed 0 matches the audited run's loss curve to three decimals, and bit-exact reproducibility of the SFT chain was not tested, see the determinism probe in App. E), plus *four seeded LoRA-DPO runs* built for the pre-registered DPO seed null of Sec. 4.2 (seeds 0–3 through the same `code/scripts/sft_seed.py` wrapper; LoRA r=16, α=32 on `q,k,v,o_proj` — the identical 6,422,528 trainable parameters as the SFT arm; 398 BFCL-derived preference pairs, 1 epoch = 12 optimizer steps, lr 5e-6, β=0.1, batch 4×accum 8, bf16, max length 1024; trainer `code/training/dpo_warmup.py` with the reference policy realized as the same weights under `disable_adapter()`, unit-tested in `code/tests/test_dpo_loss.py`; pair builder `code/training/build_dpo_pairs.py` run *once* with greedy decoding so all four seeds see an identical set, its output vendored at `data/bfcl_dpo_pairs.parquet`; job script `code/scripts/slurm/run_sig_dpo_seed_null.sbatch`, graphs `notes/basediff_dpo_seed{0..3}_l14to22.json`), GRPO baseline (@ step 100), GRPO seeds s1/s3/s4/s5 (@ steps 200 and 300 / 100 / 300 / 300; s1 was additionally re-run to step 300 to build the matched-step control of Sec. 4.2; s3 did not reach step 300, App. F), shaped-λ0.1 GRPO (@ step 150; accuracy reward plus an SAE-feature shaping penalty on the v0 panel, weight λ=0.1) — all GRPO runs full-parameter verl (Sheng et al., 2024), and all initialized *from the base model*: the training script prefers an SFT-warmed init only when a merged `config.json` exists, which the LoRA-adapter directory lacks. Headline SAEs: `qwen-scope-3-1.7b-base-w32k-l50` residual (d_{SAE}=32,768; src layer 14 → dst 22); scale arms: `qwen-scope-3-8b-base-w64k-l50` and the circuit-tracer Qwen3 transcoders (Hanna et al., 2025) (0.6–14B); cross-family arm: `Gemma-2-2B/-it` with `gemma-scope-2b-pt-res` (JumpReLU, 16k, l13→20). H2/H3 auxiliary probes use the frozen Pythia-70m reader with the public `pythia-70m-deduped-res-sm` SAE (Marks et al., 2025) and its v0 panel as labelled in the text (they read generated text, not the audited model's activations).

H1 (concentration + base control) protocol. Co-activation graph from n=500 pile-10k prompts (seq len 64, last-position states), per-state top-2,000 products, then the global top-10,000 edges; top-1%-of-active inbound-weight share, Gini, top-32 hub Jaccard, on every checkpoint with the same base-trained SAE. Degree-preserving permutation null with 3,000 permutations (destinations and the weight multiset kept, weight-to-edge assignment shuffled).

Activation battery and displacement protocols. Per-checkpoint SAE encodings over all token positions of the same n=500 probe set (plus a GSM8K variant) yield mean L_0, dead fraction, firing-frequency and mean-activation Ginis, usage entropy, and activation-intensity levels; displacement is scored as edge-set Jaccard / weighted edge cosine between 10,000-edge graphs, with the seed-noise null taken over all 15 pairs of the six GRPO-family runs.

Random-dictionary control protocol. The control of Table 3 replaces the trained `qwen-scope` encoder with a random-direction encoder: d_{SAE} = 32,768 random unit encoder directions (Gaussian, row-normalized; seed 0), with sparsity matched by top-k on the pre-activation, k calibrated to the trained SAE's mean L_0 on the same probe states ($\approx$ 50). The identical graph-building and displacement pipeline is then re-run for base, SFT, and the six GRPO-family runs on the pile protocol, and the 15-pair seed null is recomputed under the random dictionary. Receipt: `notes/circuit_rewiring_randsae.json`.

H2 (generation-side activation delta) protocol. Each audited checkpoint (SFT pre, GRPO post) generates a continuation for each of 16 probe prompts using its *own* tokenizer; the generated text is re-encoded with the Pythia-70m tokenizer, run through the frozen Pythia-70m reader, and its layer-4 residuals encoded with the public residual SAE `pythia-70m-deduped-res-sm` (Marks et al., 2025). We report per-feature mean activation over the 32-feature panel preselected by a function-word-outcome screen — features ranked by association with the next token falling in a closed set of function words, over 2,048 rollouts at sequence length 64 (positive rate 0.092); the selection receipt, including all 32 indices and the screen's definition, is committed at

Table 3: **Random-dictionary control (pile arm): a single random dictionary preserves the SFT separation and the GRPO non-detection against the GRPO-family band.** The displacement audit re-run with the trained SAE replaced by a random-direction encoder (matched $d_{\text{SAE}} = 32{,}768$, top-$k$ calibrated to the trained SAE's mean $L_0 \approx 50$); the seed null is recomputed from the same 15 GRPO-family pairs under each dictionary. One random draw (seed 0): the control shows the detection does not need the trained dictionary, not that it is invariant across random ones. Random projections churn more in absolute terms (lower null mean), but the standardized separation is preserved. Receipt: `notes/circuit_rewiring_randsae.json`.

	trained SAE	random-dir
seed null, edge Jaccard (15 pairs, mean ± SD)	0.9626 ± 0.0016	0.8802 ± 0.0025
base→SFT edge Jaccard (z)	0.937 (-15.9)	0.840 ($\mathbf{-16.4}$)
base→GRPO edge Jaccard (z)	0.962 (-0.5)	0.880 (-0.0)
base→SFT weighted edge cosine	0.9976 (n/a[‡])	0.9645 ($\mathbf{-16.9}$)

[‡] no z: the trained-SAE cosine null takes 3 distinct values over 15 pairs (SD 0.000046, below the 0.0001 reporting step) — not precision-adequate.

`notes/panel_v0_pythia70m.json` and subsequently evaluated on the v0 pilot graph in this reader space (receipt: `h2_diff_v0_cross_tokenizer.json`). *Cross-tokenizer alignment* thus happens at the text interface: each checkpoint generates with its own tokenizer, and the fixed reader guarantees a shared, checkpoint-independent feature space; the audited model's activations are never encoded by this SAE.

H3 (entropy) protocol. Same reader pipeline as H2: each checkpoint samples 4 continuations (40 new tokens, temperature 0.7) for each of 24 probe prompts (the first 24 nonempty Pile-10k rows among the first 2,000 of the `train` split); each prompt-plus-continuation sequence is decoded, truncated to 128 reader tokens, and encoded by the fixed Pythia-70m reader and SAE, giving a per-sequence mean feature activation. For each of the same 32 panel features, we normalize its per-sequence mean activation across the 96 sequences and compute the entropy of that distribution — low entropy means the feature's usage concentrates onto few sequences. Compare pre (SFT) and post (GRPO) per-feature entropies (script `sig4_h3.py`).

ICLR reproducibility checklist.

- All result artifacts are in the anonymous supplement (`notes/`): every co-activation graph (Qwen base + SFT + six GRPO-family runs + step-dose checkpoints; Gemma pair; Tulu staircase v1 and v2-normalized), the per-checkpoint activation-battery and FVU JSONs (pile, GSM8K, and on-task BFCL; App. I), the seed-null / displacement / localization / feature-interpretation summaries, the random-dictionary control (`circuit_rewiring_randsae.json` and the `randsae_*` graphs), the four DPO-seed graphs with their null receipt (`circuit_rewiring_dpo_seednull.json`) and the preference set they were trained on (`data/bfcl_dpo_pairs.parquet`), and the behavioral-calibration JSON — each headline number is traceable to a named file (the transcoder scale rows of Table 7 come from the sweep job scripts in the supplement, without a committed per-run JSON).
- Analysis scripts in the anonymous supplement (`code/` and `code/scripts/`): graph builder, battery, seed-null differencing, displacement, localization, interpretation, and log miners.
- *What is self-contained, and what is not.* The headline SFT/DPO statistics are reproducible from the supplement's graph JSONs with no GPU via `code/scripts/check_rewiring.py`. The supplement also includes the seeded-arm trainers, small input parquets, seeding wrapper, job specifications and preregistrations; rerunning training additionally requires the named public model/SAE weights, a compatible CUDA/Slurm environment and large checkpoints that are not vendored. Specifically, the training script (`code/training/sft_warmup.py`), its 900-row input (`data/bfcl_sft_train.parquet`), the seeding wrapper (`code/scripts/sft_seed.py`, which seeds before model construction so the LoRA initialization is fixed and not just the data order) and the job script (`code/scripts/slurm/run_sig_sft_seed_null.sbatch`) are all committed here, and the sbatch references no other repository. The graph builder now records the checkpoint

that produced each graph (`checkpoint`, `hf_model_path`), and we have backfilled that field for the 23 graphs feeding the headline analyses — the Qwen base/SFT/GRPO trio, the four SFT seeds, the Tulu-3 staircase, the Gemma pair, the 8B arm, and all eight random-dictionary graphs (which additionally record the random encoder, its seed and its sparsity calibration, without which a random-dictionary graph is indistinguishable from a trained-SAE graph of the same checkpoint) — using the literal `--hf-model-path` argument recorded in each committed job script (`code/scripts/stamp_graph_provenance.py`). *The remaining committed graphs, largely the 8B layer sweep, predate the field and are not stamped*; their provenance is recoverable from the job scripts but is not carried in the JSON, and we do not backfill it by inference. The four SFT-seed graphs additionally carry `training_seed` and a `training_recipe` block (LoRA configuration, optimizer settings, Slurm job id, SHA-256 of the training script, seeding wrapper and parquet). *One asymmetry we disclose rather than backfill:* the four DPO-seed graphs carry the `checkpoint`/`hf_model_path` stamp but *not* that richer `training_recipe` block, because the DPO trainer was written after the stamping helper; their recipe is recoverable from the committed job script and from the training log of JID 15102095, but it is not carried inside the JSON as the SFT-seed graphs' is. One honest limit on lineage: the *audited* SFT checkpoint was trained before we seeded the LoRA initialization, so while the four seed-null recipes and inputs are in the supplement, that original checkpoint is not re-derivable bit-for-bit — the descriptive $z = -19.9$ and range separation are reproducible from the committed graph arrays, but we do not claim bit-for-bit end-to-end training reproduction. The anonymous-bundle verifier checks its manifest, required files, core statistics and the DPO $0/4, 2/4, 4/4$ dose curve in a fresh directory. What a clean checkout does *not* contain: the two conda environments (`trl` 0.11.4/`peft` for training, `sae-lens`/`transformer-lens` for analysis), the Hugging Face caches for `Qwen3-1.7B` and the SAE weights, and the GRPO/Tulu checkpoints, which are third-party artifacts or large training outputs referenced by name rather than vendored; the training scripts that produce them are released here.

- Cross-tokenizer alignment: documented in Appendix D; the graph-differencing code is separately checked by a self-diff identity test (`h2_diff_sanity.json`: one graph against itself, Jaccard/cosine $= 1.0$), which is not itself a tokenizer test.
- Statistical reporting (H3): on the committed cross-tokenizer run (24 probes $\times$ 4 continuations, `h3_entropy_cross_tokenizer_v0.json`), mean per-feature entropy delta is -0.004 nats with Cohen's $d = -0.005$ over 32 features (the $|d| = 0.5$ entropy-collapse gate is not passed), and the hub-widening of $+0.206$ first measured on a smaller probe set does not survive — that sub-claim is withdrawn, with the discrepancy detailed in App. M (Table 14).

E Extended results: statistics, controls and calibration

This appendix holds the full text of results whose summaries appear in Sec. 4.

The calibrated estimator replicates on a second backbone and SAE family. We pre-registered the same construction on Llama-3.1-8B with Llama-Scope SAEs before any Llama quantity existed: layer pair fixed by fractional-depth formula (layers 16 $\rightarrow$ 25, as in Table 2, read at the normalization-folded operating point of App. L), identical recipe flags, four seeds, a committed reader with four branches (`PREREGISTRATION_llama_second_backbone_v2.md`; its predecessor closed attempted-and-uninformative by its own escalation clause, having produced no number). The replication branch fired: all four base$\rightarrow$SFT contrasts displace 0.0434–0.0444, every one beyond the 6-pair seed band (0.0222–0.0257); the smallest contrast is 1.7$\times$ the band's maximum. The calibrated SFT signature is not a property of one backbone or SAE family. (Receipt: `circuit_rewiring_llama_sft_seednull.json`, read only by `llama_repl_read.py`; its key `rl_signature_beyond_seed_noise` keeps the committed reader's pre-existing name and its large-effect rule (band minimum -0.02), under which it reads `false`; the registered R1 rule is band exceedance, read by `llama_repl_read.py`, and the arm is SFT.)

H1 graph statistics. Across 10,000 edges and 2,226 destination features with non-zero in-degree (the headline `qwen-scope` protocol of Sec. 3), the top 1% of active features capture 42.7% of inbound edge weight and the top 10% capture 65.0% (Table 6 and Fig. 5, App. H); the GRPO checkpoint is indistinguishable (42.9% over 2,225 active features). Both the destination-degree

distribution (max in-degree 2061) and the edge-weight magnitude (max 3597, 209× the median) are extreme-tailed: a small subset functions as a hub — and a single-checkpoint reading of either graph invites an "RL installs a reweighting signature" conclusion; the rest of this section shows why that would be wrong.

The audit's detection threshold. *The audit's detection threshold is explicit:* with null SD 0.0016, any contrast shifting edge Jaccard by $\geq$ 0.0032 — about 17 more of the 10,000 retained edges swapped out than the run-noise mean (33 more in the symmetric difference, since $J = (N - x)/(N + x)$ for x swapped edges) — is flagged at $|z| \geq 2$ (a flag threshold over 15 dependent pairs, not a power calculation at a stated α). SFT's shift is 7.9× this threshold; no RL contrast exceeds 1.0× it.

SFT against the GRPO-family band, and GRPO across steps. First, the *SFT* step displaces the graph at $z = -15.9$; because n=15 pairs cannot license a Gaussian tail we also give a descriptive range statement: base→SFT falls four times the entire null range below its minimum (rank 1 of 16, a rank rather than a tail probability; Proposition 1), while only one base→RL contrast (s1) falls below the band, by 0.21× the null range (19× less). By rank alone s1 ties SFT (both rank 1 of 16); the separation is carried by *effect size*, not band membership. Notably SFT is the *LoRA* stage (rank-16, α=32, attention projections only — 0.37% of the model) while the GRPO runs are full-parameter, so the detected displacement is not a trivial consequence of updating more parameters. That rules out a parameter-count triviality, not the algorithm-level reading: the stages also differ in objective, data and duration (Tulu-3's full-parameter SFT, which writes the staircase's largest increment but has no seed band, only partially closes the parameterization axis). Second, the seed-noise null converts "the base→GRPO change is small" into "not resolved above RL run-to-run noise" (no equivalence test; s1 falls 0.0006 below the three-pair matched band) — a non-detection even where a difference measure had the best chance of exposing one. Nor does one accumulate with training: within a single GRPO run (seed s4), base→checkpoint displacement is flat across training steps ($z = +0.3/+0.1/+0.8$ at steps 100/200/300; hub Jaccard 0.94–1.00 throughout).

The concentration is far above a degree-preserving null — but that only rules out chance, not the base model. A degree-preserving permutation null (3,000 permutations; App. D) puts the null top-1% share at $33.0\% \pm 0.5\%$; the observed 42.7% is $z = 20.0$ above it ($p \leq 1/3{,}001 \approx 0.000333$, the empirical floor of a 3,000-permutation test; $z = 10.5$ at 8B). *A trap worth naming:* the null answers "above chance?," not "did RL cause it?" — a single-checkpoint study stopping here reports a $z = 20$ effect and lets the reader infer an RL mechanism.

Which statistic we score on, and why the other one cannot carry a z. A standardized score is only meaningful if the null's dispersion exceeds the precision at which the underlying quantity is computed and reported. We commit to **edge Jaccard** as the audit's scored statistic, and to reporting the weighted edge cosine *descriptively*, because on the trained dictionary the cosine fails that test. Our similarities are recorded to four decimals; the 15-pair trained-SAE cosine null takes only *three distinct values* (0.9989, 0.9990, 0.9991) with SD 0.000046 — *below* one unit in the last recorded place. Dividing by that SD manufactures large numbers from rounding: the $z = -30.7$ a previous draft quoted for base→SFT weighted cosine is an artifact of the reporting grid, not a measured effect size, and we withdraw it. Edge Jaccard has no such problem: its null spans 14 distinct values across the 15 pairs with SD 0.0016, sixteen times the reporting step, so $z = -15.9$ (SFT) and $z = -0.5$ (GRPO) are computed against a genuinely resolved distribution. The same check passes for *both* statistics under the random dictionary (null SDs 0.0025 and 0.00057, i.e. 25 and 6 reporting steps), which is why the random-dictionary cosine $z = -16.9$ in Table 3 is quoted while its trained-SAE counterpart is not. The substantive conclusions are unchanged — they were always carried by edge Jaccard — but the discipline is the paper's own: a null-calibrated statistic must first beat its own numerical precision. The other base→RL contrasts fall at $z \in [-2.0, +0.8]$ (the one marginal case, s1 at $z = -2.0$, at matched step 300 still sits 0.0006 below the three-pair same-step band while inside the 15-pair one ($z = -0.98$), so its marginal exceedance is not a step artifact; it is 19× smaller than SFT): GRPO displaces the graph no more than one GRPO seed does relative to another — not resolved beyond run noise, which neither measures nor excludes a change shared by all runs.

Weighted edge cosine at full precision. The four-decimal limit above belongs to the stored similarity receipts, not to the graphs. Recomputing $1 -$ weighted edge cosine (cosine of $|w|$ over the union of retained edges) at full floating-point precision from the committed graphs, for the same runs and bands, gives the same outside counts as edge Jaccard for LoRA-SFT ($4/4$; band maximum 0.00106, smallest contrast 0.00228, $2.2\times$ the band maximum), the GRPO family ($1/6$) and DPO at 29 and 60 steps ($2/4$, $4/4$); at 12 steps and for the matched-construction DPO runs it puts one contrast outside where edge Jaccard puts none ($1/4$ vs. $0/4$ each). We report counts only, no z. The check was pre-registered as descriptive, with no branch (`notes/PREREG_sftband_20260926.md`, analysis R0, which names SFT, GRPO and DPO at 12 and 29 steps; DPO at 60 steps and the matched-construction runs were read in the same pass). Receipt: `results/sftband_20260926/r0_stored/r0_stored_graphs.json`; reader `code/scripts/sftband_readouts.py`, which reproduces the committed Qwen3-1.7B edge-Jaccard bands and counts of Table 1 before adding the cosine column.

A per-algorithm null for the SFT stage: the positive half, calibrated by its own standard. The audit's thesis is that a post-training-internals claim must be scored against *that algorithm's* run-to-run variability. One SFT checkpoint scored against the GRPO-family band would not meet that bar: it is a borrowed null, exactly the substitution the introduction criticizes. We therefore trained three further LoRA-SFT runs plus a re-seeded replicate of the audited recipe (four seeded runs, identical data, hyperparameters and step count; App. D) and built the SFT stage's own 6-pair run null. These six distances share four run endpoints, so z is a descriptive distance scale, not a calibrated tail statistic; the four-run range separation and cross-backbone replication carry the inference. It is *tighter* than the GRPO one — SFT runs resemble each other slightly more than GRPO runs do (SD 0.00137 vs. 0.00160; mean 0.9644 vs. 0.9626; band $[0.9619, 0.9656]$, six distinct values, SD $14\times$ the reporting step) — so the detection gets *stronger*, not weaker, when the borrowed null is replaced by the right one: base→SFT moves from $z = -15.9$ to $\mathbf{z = -19.9}$, falling $6.7\times$ the null range below the most extreme null pair (against $4.0\times$ before). The result is not a property of the one audited checkpoint: all five base→SFT contrasts land at $z \in [-20.0, -17.0]$ (0.937–0.941), every one of them below the SFT seed band, while the six seed pairs stay inside it. The stage that writes a footprint does so reproducibly, and the stage that does not (GRPO, $z = -0.5$) is now compared against a like-for-like construction. Receipt: `notes/circuit_rewiring_sft_seednull.json`. (Consistent with Sec. 3's precision rule, the weighted cosine again fails the precision-adequacy check on this null — SD 0.000041, below the reporting step — and we quote no z for it.)

A per-algorithm null for DPO: the same construction, returning a negative at the registered 12-step dose. The asymmetry of calibrating SFT and GRPO against their own algorithms but DPO against neither is closed here by measurement rather than by scoping. *The design was fixed first.* `PREREGISTRATION_dpo_seed_null.md` was committed before any DPO seed was trained: recipe, seeds, null construction, contrast, a <200-pair data floor, and four exhaustive interpretation branches (D1 separates, D2 does not, D3 degenerate band, D4 under-powered data). Git history is the timestamp, and `notes/circuit_rewiring_dpo_seednull.json` did not exist when it was written.

The null. Four LoRA-DPO runs (seeds 0–3) through the same pre-model-construction seeding wrapper the SFT null used, on 398 BFCL-derived preference pairs, give the same 6-pair null over 4 runs: mean 0.96705, SD 0.002202, band $[0.9643, 0.9706]$, six distinct values, SD $22\times$ the reporting step — so the statistic is precision-adequate here as it is for SFT and GRPO.

The 12-step verdict is D2, and the D3 positive control is what makes it readable. A null is worth nothing if the arm could not have detected anything, so the registration required the DPO runs to displace the graph by more than the SFT seed band's width or be declared unable to resolve. They clear it comfortably: base→DPO displacement $(1 - J)$ is $0.0352/0.0354/0.0352/0.0327$ across the four seeds, i.e. 6.1–$6.6\times$ the width of a 95% interval on the SFT seed null ($3.92 \times 0.001367 = 0.00536$) and 8.8–$9.6\times$ its literal min–max width ($0.9656 - 0.9619 = 0.0037$). We quote the 95% width because it is the more conservative reading of "the SFT seed band's width" as registered; on the min–max reading the control clears by more, not less. The recipe trained and the instrument can resolve — and at this dose *every* base→DPO contrast falls *inside* the DPO band ($z = -1.02$, -1.11, -1.02, $+0.11$; the scored run is seed 0 at $z = -1.02$). Put the two stages side by side on the same axis: SFT's base-to-post change is $1.8\times$ its own seed-to-seed change (0.0628 vs. 0.0356),

DPO's is $1.07\times$ (0.0352 vs. 0.03295). DPO displaces 3.5% of a 10,000-edge graph and that is, on this recipe, what two runs of it do to each other. At this registered dose, this is the paper's thesis meeting its sharpest case: a difference can be genuinely large and still not be resolved from run-to-run noise, and only the null tells the two apart. The weighted edge cosine agrees in sign and magnitude ($z = -1.11$) but its null SD is 0.000075, below the reporting step, so by the precision rule of Sec. 3 it is reported descriptively and carries no part of the claim.

What this does not license. Not "preference learning leaves no trace." The claim is scoped to LoRA-DPO (r=16, α=32, attention projections) on `Qwen3-1.7B`, on 398 BFCL-derived preference pairs for 12 optimizer steps, read on general-text probe states through one co-activation statistic. Three bounds in particular. *(i) Dose.* The 12-step D2 verdict does not persist as a band-membership statement: the registered ladder below gives $2/4$ contrasts outside at 29 steps and $4/4$ at 60, while absolute displacement remains flat. *(ii) Scale and parameterization.* 1.7B, LoRA, one preference set. *(iii) It does not transfer to Tulu-3.* The public staircase's DPO stage is full-parameter DPO on a 271k-pair mixture; our 1.7B LoRA run is a different procedure, and reading its seed noise onto that stage would be exactly the borrowed-null substitution this paper exists to criticize. The Tulu-3 DPO increment therefore keeps the RLVR run-pair reference it has (Sec. 4.3, L8). Receipt: `notes/circuit_rewiring_dpo_seednull.json`; graphs `notes/basediff_dpo_seed{0..3}_l14to22.json`; job script `code/scripts/slurm/run_sig_dpo_seed_null.sbatch`.

The dose-matched ladder: separating dose from objective (L8a), and what it costs. The 12-step arm above confounds objective with dose — it is 12 optimizer steps against the SFT arm's 29. We registered the fix before running it (`PREREGISTRATION_dpo_dose_matched.md`, committed at 01:05:07 with the job starting 01:05:10; a three-second margin, which we state rather than round up) and retrained all four seeds on a single 60-step trajectory with adapters at the two registered rungs: 29 steps (SFT's exact count, the primary) and 60 ($\approx 2\times$). We chose step-matching over displacement-matching in advance because matching displacement requires building a graph, reading the number, and deciding whether to keep training — an adaptive stopping rule on the same statistic that carries the verdict. Job 15103738; receipts `notes/circuit_rewiring_dpo_dose{29,60}.json`.

The result that needs no null. At the identical 29 optimizer steps the four SFT runs displace the graph by 0.0588–0.0630 and the four DPO runs by 0.0323–0.0365. The ranges are *disjoint*, separated by a gap (0.0223) $5.3\times$ either range's own width (0.0042 for both arms), so SFT's mean displacement is $1.78\times$ DPO's at equal dose. (We quote the ratio of means, not of extremes: the extreme ratio spans 1.61–1.95 and its upper end sits on a rounding boundary, so it is not a stable number to report.) This comparison invokes no seed band, no z, and no assumption about reproducibility; it is the strongest thing the ladder buys. And dose does not rescue the gap: DPO's displacement is flat across the whole $5\times$ ladder (4-run means 0.0346/0.0343/0.0349 at 12/29/60, while the seed-pair mean μ_{seed} rises 0.9671/0.9677/0.9692 and σ_{seed} falls 0.0022/0.0012/0.0012; all twelve contrasts in 0.0323–0.0365), never approaching SFT's 0.0628. We say "flat over this range" and not "saturates": three rungs in a narrow band support the absence of growth, not convergence to a limit.

The registered checks (numbered as registered). (1) The D3-style positive control (a manipulation check that DPO moved the graph at all, distinct from the instrument-level positive control of Sec. 1) passes at every rung (6.0–$6.8\times$ the SFT band's 95% width). (4) Edge Jaccard is precision-adequate at every rung (null SD 22/12/12 reporting steps); the weighted cosine fails everywhere and carries nothing. (3) Monotonicity *fails* — displacement does not grow with dose — which the registration required us to state outright rather than read a null off a run that had stopped moving; that non-growth is the finding of the previous paragraph. (2) The trajectory-continuity check *fails*, and the registered response (branch M5) is to report the rungs as three independent arms at three doses rather than as one dose-response curve, which is what we do.

Why continuity failed, diagnosed rather than assumed. Steps 1–12 of the ladder runs do not reproduce the registered 12-step runs (max $|\Delta\text{loss}| = 0.064$, diverging at step 2; these loss traces are recorded in the lab notebook, not in a committed JSON). We ran a probe (job 15109030) that trains seed 0 twice on one node under the old and new code paths. The two agree exactly for five optimizer steps and then diverge, on provably identical batches — as does a fresh run of the *unmodified* path against the registered one on a different node of the same GPU model. So the divergence is neither

our `--max-steps` edit nor the A100→H100 move: **this training is not bitwise reproducible at fixed seed, code and hardware**, presumably through nondeterministic bf16 reductions, and the registered check assumed a determinism the system does not have. That is a defect in our registration, not a finding about DPO. It has one benign consequence — each null is then a genuine *run-to-run* null, not merely an initialization null, which is what "two runs of the same algorithm" should mean — and one warning: no claim in this paper should be read as asserting bit-exact reproducibility of the DPO chain. (The SFT chain is untested by this probe; its reported claim is a three-decimal loss-curve match, which this does not contradict.)

Two ways of reading the band, which disagree, so we report both. The `rl_signature_beyond_seed_noise` field in every receipt fires on `base→focus` $<$ band minimum $- 0.02$ — a *large-effect* rule, not band membership. Under it no DPO rung separates (SFT clears it at a margin of 0.0247; no DPO rung comes within 0.019). Under the rule our own SFT paragraph states in prose — every contrast outside the band while every seed pair stays inside — the rungs disagree: $0/4$ contrasts outside at dose 12, $2/4$ at 29, $4/4$ at 60 (per-contrast z ranges $[-1.11, +0.11]$, $[-3.61, -0.01]$, $[-4.10, -2.56]$; all six seed pairs stay inside at every rung). Registered branch M3 covers exactly this case and prescribes reporting the curve and promoting no single z; being the more conservative of the two readings that have support, that is what we do. *The arithmetic of the turn matters:* it is driven by the band, not the contrasts. Displacement is flat, while the seed band rises with dose (enlarging $\mu_{\text{seed}} - c_s$) while its spread narrows from 12 to 29 steps and then holds (mean $0.96705 \to 0.96772 \to 0.96917$, SD $0.0022 \to 0.0012 \to 0.0012$), because the four seeds converge toward each other as they train. A null that moves while the effect stands still will eventually separate anything, which is why we decline to read dose 60 as a detection.

One asymmetry between our two nulls, disclosed because it cuts the wrong way. `dpo_warmup.py` uses a fixed, unshuffled index order identical across seeds (registered as a known property), so the DPO null varies by initialization and execution nondeterminism but *not* by data order. The SFT arm runs through `trl`'s `SFTTrainer` with `seed` set in `SFTConfig`, so its null *does* include data-order variance. At dose 12 this favoured nothing (the DPO band was the wider of the two, SD 0.0022 vs. 0.00137); by dose 29–60 the initialization differences wash out and the DPO band tightens below SFT's, partly because a variance component is structurally absent from it. That is *anti*-conservative for detecting DPO, and it is the second reason we do not promote the dose-60 reading. A matched-construction DPO null — per-run seeded reshuffling — is the specific fix; we pre-registered and ran it (`PREREGISTRATION_dpo_matched_construction.md`): four fresh 29-step runs land at 0.0330–0.0356, inside the fixed-order 29-step contrast range (0.0323–0.0365) and inside their own reshuffled band ($0/4$ outside; SD 0.0019 vs. 0.0012 fixed-order), and still disjoint from SFT's 0.0588–0.0630; the gap to SFT is $\sim 9\times$ the MC contrast range (0.0232 vs. 0.0026). The data-order concern therefore retires for the disjointness comparison, deliberately not for the band reading: with data-order variance included the 29-step reading is $0/4$ rather than $2/4$, and dose 60 was not rerun this way. Receipt: `circuit_rewiring_dpo_mc29.json`, read only by the committed `mc_null_read.py`.

Two disclosures the DPO arm requires. *(1) The preference set was rebuilt after the registration, and the reason is recorded in it (AMENDMENT 1).* The first build satisfied the registered condition — "rejected" strings string-unequal to the gold call — at $900/900$ rows and was nevertheless invalid for the registered purpose: *every* rejected string was a Qwen3 `<think>` monologue (mean 400 characters against the gold call's 76; 0 of 900 parsed as a function call), so the preference contrast was **format** (concise call vs. reasoning block), not **correctness**. A DPO null trained on it would have measured what suppressing the think-block does to activation geometry, under a valid pre-registration timestamp. We fixed it at the source rather than by post-filtering: generations now use `apply_chat_template(..., enable_thinking=False)`, any generation that still does not parse as `name(` is dropped and counted separately as a format contrast, and `</think>` residue is stripped before comparison. The rebuilt set is 398 usable pairs of 900 rows — 273 dropped because the base model already emits the gold call (no pair exists) and 229 dropped as non-calls — clearing the registered 200-pair floor. The amendment was written before any seed was trained, no DPO co-activation graph existed, and the invalid parquet was deleted rather than kept as a fallback. The general lesson is worth stating: a registration can be satisfied to the letter by data that answers a different question, and the arithmetic that exposed it here ("0 of 900 rejected strings parse as a call") was not in the registration because it did not occur to us to require it.

(2) The DPO trainer is ours, not `trl`'s. `trl` 0.11.4's `DPOTrainer` is incompatible with the `transformers` 4.57.3 in this environment (`get_batch_samples() takes 3 positional arguments but 4 were given`; JID 15099744), so rather than chase versions we implemented the objective directly in `code/training/dpo_warmup.py`: $\mathcal{L} = -\log\sigma(\beta[(\log\pi_\theta(y_c) - \log\pi_{\text{ref}}(y_c)) - (\log\pi_\theta(y_r) - \log\pi_{\text{ref}}(y_r))])$ with β=0.1, completion-only masking, and the reference model realized as the *same weights with the LoRA adapter disabled* — so the reference is provably the base policy and no second model is loaded. Because a hand-written DPO can be wrong in two ways that still train and still yield a seed null of the wrong quantity, both are unit-tested against hand-computed values in `code/tests/test_dpo_loss.py` (5 checks): prompt tokens leaking into the sequence log-probability (masking, padding-invariance) and an inverted preference sign (prefer-chosen $<$ neutral $<$ prefer-rejected, with neutral $= \log 2 = 0.6931$ exactly). The measured loss is interpretable against that constant: seed 0 runs $0.6641 \to 0.6356$ and seeds 1–3 start at 0.6931 and end 0.6198/0.6527/0.6295, i.e. every run moves off neutral in the direction of preferring the chosen completion. Writing the loss out also had the side benefit of matching `sft_warmup.py`'s data path and LoRA construction exactly, which the `trl` route did not.

Random-dictionary control, in full. Trained-SAE evaluations often fail to beat random baselines — random feature directions match trained dictionaries on interpretability, sparse probing, and causal editing (Korznikov et al., 2026) — so a detection claim made *through* a trained dictionary must run that gate on itself. We re-ran the entire pile-arm displacement audit with the trained SAE replaced by a *random-direction* encoder — random unit encoder directions at matched width ($d_{\text{SAE}} = 32{,}768$), sparsity matched by top-k calibrated to the trained SAE's mean L_0 (≈ 50) — rebuilding all eight graphs and recomputing the 15-pair seed null under the random dictionary (protocol in App. D). The result (Table 3): the SFT detection *fully survives* ($z = -16.4$ random vs. -15.9 trained on edge Jaccard; $z = -16.9$ on the random dictionary's weighted cosine, whose trained-SAE counterpart is too quantized to score) and the RL null holds under both ($z = -0.0$ random, -0.5 trained). Two consequences, one deflationary and one useful. Deflationary: the detection is *not* evidence about trained SAE features — the single matched random projection tested here reads it out as well, which is why we do not brand it an "SAE signature." Useful: the detector does not need the trained dictionary, so the seed-noise-null audit is cheap. *We state this control's scope exactly, because it is one run:* it draws a *single* random dictionary (seed 0; App. D), so it licenses "the detection does not require the *trained* dictionary," not "the detection is invariant to the choice of projection" — that would need a null over projection seeds, which we did not run. Only the *interpretive* layer (naming the displaced hub features, App. J) requires the trained SAE, and this control cannot certify that layer.

Behavioral sanity check, in full (provenance and intervals). We evaluate the checkpoints with the verl `val_only` harness on the held-out BFCL validation split (SFT LoRA merged before evaluation), drawn from the same release files and categories as the training rows (App. D). Accuracy is the unweighted mean of the four category accuracies on 100 held-out items (40/20/20/20), and each 95% CI is a stratified normal approximation, $\text{Var} = \frac{1}{16}\sum_c p_c(1-p_c)/n_c$. The base model scores 0.531 ($[0.43, 0.63]$); SFT reaches 0.6375 $[0.54, 0.73]$; the audited GRPO checkpoint reaches 0.8875–0.900 $[0.79, 1.00]$ (lower: independent re-evaluation of the audited weights, upper: in-training validation; our receipts keep only the category mean for GRPO (the per-category values were not extracted from the verl logs), so its interval uses the bound $p_c(1-p_c) \le 1/4$; $[0.79, 1.00]$ spans the intervals of the two GRPO readings, its upper end truncated at 1). The two gains are not on equal statistical footing: GRPO's is unambiguous (conservative unpaired $\Delta = +0.356$, SE 0.072, 95% CI $[+0.21, +0.50]$) while SFT's is *not* resolved by this harness ($\Delta = +0.106$, SE 0.070, 95% CI $[-0.03, +0.24]$, which includes zero). *What this measurement is for, and what it is not.* It exists to defeat one objection — that the checkpoint reading flat internally simply never learned anything — and GRPO's interval settles that. We draw *no* behavioral/internal dissociation from it. A withdrawn earlier draft reported the contrast as a headline dissociation and quoted SFT's gain as $3\times$ smaller; both claims are superseded because the ratio divides a solid estimate by one indistinguishable from zero, and the two arms are measured on different state distributions (accuracy on BFCL task states, displacement on general text) with no calibrated on-task internal instrument to pair them, since the matched on-task audit fails its own positive control (SFT $z = -1.06$ on-task vs. -19.9 on pile). *Provenance:* the harness is the RL task's own held-out validation split of 100 items (40/20/20/20 over the four training categories); the CIs are unpaired, which is conservative if an item's correctness

is non-negatively correlated across checkpoints, since all checkpoints are scored on the same 100 items but we hold no per-item records to pair on. The split sizes follow from the split script and are confirmed by the per-category accuracies (e.g. base `simple_python` $0.575 = 23/40$). The three scores are *not* from a single job: base and SFT come from JID 12408266 and the GRPO merge-eval from JID 8439216, the same verl `val_only` configuration run at different times, so an unmeasured harness-version difference is not excluded. A single re-scoring job producing per-item outputs and paired intervals is the clean fix and is listed as open work.

On-task arm, in full (see also App. I). Re-running the full audit on BFCL task states (n=500 chat-templated training prompts, last-token decision points, same protocol; App. I) was intended to close the cross-state gap. It cannot: the on-task instrument is structurally uninformative at the relevant effect size. Its 2σ minimum detectable effect on edge Jaccard is 0.0806 — 10.7% of the statistic, against SFT's actual pile-arm effect of 0.0254 (2.6%) — so the arm can only detect effects $\sim 3\times$ larger in absolute Jaccard ($\sim 4\times$ relative to each arm's statistic) than the one the audit's positive control produces, and the SFT positive control indeed does not separate on it ($z = -1.06$ vs. pile -15.9). We therefore report the arm's outcomes descriptively (base→GRPO $z = -0.23$; all seven post checkpoints $|\Delta\text{Gini}| \leq 0.026$; battery within seed noise on all eight informative metrics, the ninth being the TopK-pinned mean L_0; validation-split trio flat) together with its MDE, and draw *no conclusion from it in either direction*: the RL-null claim lives on the pile arm only (App. I quantifies this, including the arm's contaminated null band).

The null is not basis-blindness: the delta itself is measured. A remaining escape route is that the RL footprint hides where the SAE cannot see — e.g. in the $\sim 6\%$ reconstruction-error term the coarse FVE bound cannot exclude. We close this by measuring, per token, the true activation change Δh and its SAE-represented counterpart $\Delta\hat{h} = \text{SAE}(h_{\text{post}}) - \text{SAE}(h_{\text{base}})$ (n=200 prompts, 12.8k tokens). GRPO *does* move activations ($\overline{\|\Delta h\|} = 25.5$; SFT $1.6\times$ more, 41.9 — the same ordering as every other measure), and the code over-responds rather than misses ($\|\Delta\hat{h}\|^2/\|\Delta h\|^2 = 3.0$ for GRPO, 1.9 for SFT; cosine $0.26/0.32$; even the 10th-percentile token retains $\geq 21\%$ of its delta energy in-code) — the signature of sparse thresholded codes. These checks rule out *complete* SAE insensitivity to the RL delta — the code responds, and over-responds, rather than missing it — but the moderate delta-cosine ($0.26/0.32$) means the SAE captures the difference *direction* only partially, so difference-specific blind spots cannot be fully excluded. The SAE churns under the RL delta at roughly the rate two RL seeds churn against each other, which is why every seed-nulled statistic reads null. With the FVE bound, the insensitivity objection is bounded from both ends.

Localizing the SFT signature (App. J). Having *detected* SFT displacement, the same machinery names *where* it acts: 63% of the edge weight SFT drops (but base and all six GRPO-family runs retain) is inbound to a single destination — #30329, the rank-1 concentration hub — a localized pruning of boilerplate/function-word wiring that GRPO leaves intact. A causal ablation probe keeps the claim honest: the interaction is real and feature-specific, but the edge-drop is a change in *co-activation statistics*, not (by this probe) a pruned causal coupling — strictly correlational. This interpretive layer is also the only part of the audit that requires the trained SAE: the random-dictionary control (Table 3) certifies the detection but has no feature identities to offer, so the hub-level naming rests on the trained dictionary and cannot be certified by that control.

The SAE blind-spot bound, in full. The FVU blind-spot bound lives on the `qwen-scope` family, where the decode path is verified — and it is tight: the base-trained SAE reconstructs base, SFT, and GRPO equally well (FVU $0.0592/0.0588/0.0593$, FVE $\approx 94.1\%$). Had post-training moved activations into directions the base SAE cannot represent, post-checkpoint FVU would rise; it does not: 94% of variance stays on the base basis at every stage — the nulls are not an artifact of a *wholly* blind basis (this marginal bound does not by itself exclude difference-specific blind spots; those are addressed by the delta projection above).

Operating-point robustness, in full (see also App. L). The staircase was run at both `Llama-Scope` normalization operating points and every conclusion is robust (the post-SFT stage increments agree to within 0.001; the base→SFT increment, the largest-displacement stage, shifts $0.669 \to 0.697$ but stays by far the dominant stage; DPO→PPO straddles a rounding boundary, v1

0.958 / v2 0.959); absolute L_0 is never compared across SAE families, and `Llama-Scope` FVU is excluded (verified decode-path loader inconsistency).

F FULL LIMITATIONS AND UNABRIDGED BODY PARAGRAPHS

Limitations (full text). (L1) The decisive *light*-RL base/SFT/GRPO control is at 1.7B, replicated across five Qwen runs and a second family; the one cross-family *post* checkpoint (`gemma-2-2b-it`) is heavy production instruct-tuning, not light GRPO, so it bounds "heavy post-training," not "light GRPO"; no same-backbone 8B light-GRPO trio exists in `qwen-scope`, so 8B-scale generalization is carried by the Tulu-3 staircase (Sec. 4.3), whose RL stage is production RLVR. Long-horizon math-RLVR at larger scale sits outside this envelope entirely: nothing here bounds RL's internal footprint in that regime, and the RL non-detection should not be read as a claim about it. (L2) The top-32 panel is H1-selected, not random. (L3) Cross-tokenizer alignment has its own noise floor. (L4) The dictionary-relative "top-1%-of-d_{SAE}" statistic (84.0% at 8B) is not comparable across SAEs of different width (cf. the width dependence of SAE statistics in Zaher et al., 2026); the 8B coincidence (#35239) is a shared *base-model* hub pending a same-backbone 8B base/GRPO check. (L5) The on-task (BFCL) arm (Sec. 4.2; App. I) is reported as *uninformative*, not as a null: its 2σ MDE (0.0806 edge Jaccard, 10.7% of the statistic) is $\sim 3\times$ SFT's actual pile-arm effect (0.0254, 2.6%; $\sim 4\times$ in relative terms), and the SFT positive control does not separate on it ($z = -1.06$ vs. pile -15.9); we state its MDE and draw no conclusion from it, so the RL-null claim rests on the pile arm alone. (L6) A fourth staircase on OLMo-2 (public base→SFT→DPO→RLVR chain, 60-step RLVR branches) is *blocked, not omitted*: no base-trained OLMo-2 SAE weights are public (the one candidate repository holds interpretation exports, not weights), so that audit first requires training an SAE — future work. (L7) The seed-noise band conflates seed, checkpoint-step (100–300), and reward-recipe (one shaped-λ0.1 run) variance: of the six runs entering the 15-pair null, the only same-recipe, same-step pair is s4↔s5@300, so the band is wider than pure seed noise — conservative for detection, *anti*-conservative for certifying the RL null. The contamination is directly visible on-task, where base→s4 lands at $z = +1.83$, *above* the band's upper edge — a base→RL contrast more base-like than the seed pairs are to each other. The pile-arm matched-step control (s1/s4/s5@300, Sec. 4.2) is the partial remedy; a $\geq$ 3-seed same-recipe, same-step null on every arm is the full one. (L8) *Null coverage is uniform across the three algorithms we train, and the residual limits are about dose and transfer.* SFT has its own 4-run seed null, GRPO a 6-run one, and DPO a pre-registered 4-run one. At the registered 12 steps no DPO contrast leaves its band ($z = -1.02$ for the scored run against mean 0.96705, SD 0.002202); at 29 and 60 steps $2/4$ and $4/4$ do while displacement stays flat, so DPO's band membership is dose-dependent and the title claims only SFT. Four bounds on that result. *(a) The step-count difference between the arms is removed; the recipes still differ.* The registered arm is 12 optimizer steps on 398 pairs against the SFT arm's 29 on 900 rows, so "DPO writes less footprint than SFT" was not separable from "this DPO run was shorter." A pre-registered dose ladder addresses the step count: at the *identical* 29 steps the four SFT and four DPO runs occupy disjoint displacement ranges (0.0588–0.0630 vs. 0.0323–0.0365; SFT's mean $1.78\times$ DPO's), and DPO's displacement is flat across a $5\times$ dose range, so "still too short" is not available as an objection. What remains is a difference in training data (398 pairs vs. 900 rows), in *learning rate* (5e-6 vs. 5e-5; the $10\times$ ratio was fixed in the registration, not tuned) and in LoRA dropout (DPO 0.0, SFT 0.05) and the fact that the band reading is dose-dependent while the displacement comparison is not. *(b) The claim is one recipe at one scale:* LoRA-DPO on `Qwen3-1.7B` with a BFCL-derived preference set; it licenses nothing about preference learning in general. *(c) It does not transfer to the public staircase:* the Tulu-3 DPO stage is full-parameter DPO on a 271k-pair mixture and still supplies a *single DPO checkpoint*, so its increment (0.796) is still read against the two RLVR runs' run-pair null (0.903) — a single descriptive reference that gives no guaranteed bound on run noise and is not DPO's own. Substituting our 1.7B LoRA null for it would be the borrowed-null error this paper exists to criticize. *(d) The preference data was rebuilt mid-arm* after the first build proved to encode a format rather than a correctness contrast (App. E, AMENDMENT 1); the registered floor and the amendment are both on the record, but the arm is one build away from its registration. (L9) The behavioral calibration is n=100 (40/20/20/20 over four categories, unweighted category mean) and comes from two separate evaluation jobs; the SFT behavioral gain is not resolved from zero on it (Fig. 4, App. E).

Domain robustness: the null holds on a second probe domain. Re-running the battery on the same trio with GSM8K question+solution chains (n=500) — far more concentrated than pile (base firing-freq Gini 0.840 vs. 0.683), a genuinely different region of the basis — leaves the base→GRPO deltas nil (every base→GRPO metric |rel Δ| $< 1\%$; App. H). *Scope:* GSM8K is still *off*-training-distribution; the on-distribution (BFCL) audit is in App. I.

Do our top-k conventions select the answer? Three statistics in this paper select a top-k: the 10,000 retained edges, the 32 destination hubs, and the random-encoder top-k. Two exposures follow. *(i) Tie-breaking.* `argsort` and `topk` order ties by index, so over sparse SAE values a slot of exactly zero would be ranked by feature id. We check the committed graphs: no exact-zero weight survives into the retained edge set, and the 32nd and 33rd hubs never tie, so no hub Jaccard in this paper turns on an index-ordered tie-break. One residual is disclosed: the smallest retained weight has multiplicity up to 5, and the committed graphs do not contain the 10,001st edge, so ties exactly at the cutoff cannot be bounded from them. *(ii) Does the convention choose the conclusion?* Recomputing the SFT seed null over top-2,000 and top-5,000 subsets of the same graphs leaves SFT detected at every N ($z = -14.6$, -14.9, -20.0 at $N = 2{,}000/5{,}000/10{,}000$; computed from unrounded similarities, whereas the four-decimal receipt gives the -19.9 quoted elsewhere at $N = 10{,}000$). The cutoff moves the magnitude but not the conclusion, so we quote N alongside the statistic. Receipt: `code/scripts/audit_conventions.py`.

Why the *public* DPO stage still has no per-algorithm null: the search and the cost. *Superseded in part:* an earlier draft used this paragraph to argue that no DPO seed null should be run at all. That argument is withdrawn as to our own arm — the pre-registered 1.7B LoRA-DPO null above was run, and it answers a well-posed question — but its core reasoning was right and still stands as to the *public* stage, so we keep it here rather than delete it. The public Tulu-3 staircase provides exactly one DPO policy checkpoint, so the DPO stage increment (0.796) is scored against the two RLVR runs' pair null (0.903) — a single descriptive reference that also varies algorithm and recipe, gives no guaranteed bound on same-recipe run noise, and is not DPO's own. We checked whether a second DPO policy exists publicly: it does not. The only candidate, `Llama-3.1-Tulu-3-8B-DPO-RM-RB2`, is a *reward model* (a sequence classifier released with RewardBench 2) built on top of the DPO checkpoint, not a second DPO policy branched from the same SFT parent. Replicating the stage ourselves remains out of budget: Tulu-3's DPO is *full-parameter* DPO on a 271k-pair preference mixture, so 2–3 seeds of it is tens of GPU-hours on multi-GPU nodes, more than the $\sim$35 GPU-hours estimated for all results here. And the cheap substitute is not a substitute: our LoRA-DPO run at 1.7B measures the run noise of a *different* procedure, so it is reported as its own result about its own recipe and is *not* transferred onto the Tulu-3 stage. Doing that would be precisely the borrowed-null error this paper exists to criticize. Consequently the title claims only SFT; the Tulu-3 DPO increment is reported descriptively against one non-matched RLVR pair; and a seeded null on the public recipe, at production scale, is still the experiment that would calibrate it.

Seed-null band accounting, Tulu staircase and discussion. The GRPO-family comparison is in Table 9 and Fig. 6a. Alongside the band-exceedance count K_{out} of Sec. 3, we give a descriptive standardized score against the 15 GRPO-family pairwise similarities (mean 0.9626, SD 0.0016) and how far a contrast falls below the most extreme null pair, in units of the null range. (The 15 pairs share six underlying runs and are not independent; we therefore lean on the descriptive rank/range statements and quote z as an effect size, not a calibrated tail probability.) *An honest accounting of the band itself:* the null mixes seeds, checkpoint steps, and one shaped-reward recipe, which widens it — conservative for detection but *anti*-conservative for certifying the RL null (a wider band is easier to sit inside). The matched same-recipe, same-step (@300) seed null now spans *three* pairs — s1, s4, s5 all at step 300 (s1 re-run to the matched step for this control) — with seed-to-seed Jaccard {s1↔s4 0.9616, s1↔s5 0.9621, s4↔s5 0.9644}, a band $[0.9616, 0.9644]$ (mean 0.9627); the three matched base→post@300 contrasts are base→s1 = 0.9610, base→s4 = 0.9639, base→s5 = 0.9631 (mean 0.9627): two lie inside the band and one (s1) lies 0.0006 below its lower edge — a gap smaller than the band's own width (0.0028) — and the two group means are equal to four decimals (0.9627 vs. 0.9627). At matched recipe and step, then, the two means agree to four decimals — descriptively consistent with the base-to-RL change being the same scale as seed-to-seed run noise. We deliberately claim no more: the three seed pairs come

from only three runs and the three base contrasts all share one base checkpoint, so these are dependent observations, not six independent replicates, and no equivalence margin is applied. This is a descriptive consistency, not a test of equivalence, and not a certification of the RL null. The fourth seed (s3) did not reach step 300, so under branch C of the pre-registered equivalence analysis (`preregistrations/PREREG_matched_step300_equivalence.md` in the supplement, whose margin, statistic and branches were committed before any four-run matched graph existed) we report this three-pair result as it stands, make no equivalence claim, and record the shortfall. The SFT detection, whose z only shrinks under a widened band, is unaffected. The base→GRPO edge Jaccard is 0.962: $z = -0.5$, rank quantile 0.38 — dead center of the seed null; the weighted edge cosine (0.999) is identical to the seed-pair cosine to three decimals. *We name this checkpoint's step, because it is the shortest run in the audit:* the "base→GRPO" headline is the accuracy-only GRPO baseline at `global_step_100`, whereas the GRPO runs reach 300 steps. The null does not depend on reading the shortest run: the two seeds carried to step 300 under the identical recipe give base→s4 $z = +0.84$ and base→s5 $z = +0.34$ — if anything *more* base-like than the seed pairs are to each other — and s1@300 gives $z = -0.98$, inside the band. All three endpoint contrasts sit inside the seed band, so the step-100 choice is not what produces the null — but we should not overstate the uniformity: s1@300 ($z = -0.98$) is in fact farther from zero than the step-100 headline ($z = -0.5$), so "every endpoint is at least as null" would be false as written. The weaker, sufficient claim is what we assert: no endpoint contrast escapes the band.

Because the Qwen arm is limited to *light* GRPO, we run the full audit on a public production post-training *staircase* (Lambert et al., 2025): `Llama-3.1-8B` base (Grattafiori et al., 2024) → `Tulu-3-SFT` → `Tulu-3-DPO` (Rafailov et al., 2023) → `Tulu-3-8B` (RLVR, PPO (Schulman et al., 2017)), plus `Tulu-3.1-8B`, a *second* RLVR run (GRPO, improved recipe) branched from the same DPO parent — a public "run-pair null." Because the two runs also differ in algorithm and recipe, this run-pair null is expected to exceed pure run noise, though a single pair guarantees no such bound — conservative for *detecting* SFT/DPO-scale effects, *anti*-conservative for *certifying* the RL null (DPO→GRPO 0.906 vs. the 0.903 pair value is razor-thin); the RL-null claim therefore rests on the Qwen seed null, with Tulu as the production-scale consistency check, not an independent certification. SAEs: base-trained `Llama-Scope` (He et al., 2024) residual SAEs (l16→25, 131k, matched relative depth). Three findings (Table 2; App. Fig. 7).

(1) Levels stay base-set even at production scale. The base model's Gini is 0.780, and every post-training stage sits within $+0.009$ of it (v1; ≤ 0.019 at the normalized point, App. L); the activation battery agrees (mean L_0 -0.6%, firing-frequency Gini -0.9%, usage entropy unchanged) — the level nulls replicate in a third family and SAE basis, at production intensity. `Llama-Scope` is JumpReLU, so unlike the Qwen arm its L_0 is data-dependent and free to move ($1409.4 \rightarrow 1401.3$ across the staircase): this is one of the two arms where the sparsification metric is an actual measurement. **(2) Differences *do* detect production post-training.** Cumulative displacement against base is large — edge Jaccard 0.60–0.67, with top-32 hub turnover $0.73 \rightarrow 0.56$ — and far beyond the run-pair null of 0.90. **(3) Stage decomposition: no RLVR increment is resolved above the single available reference.** SFT displaces most, DPO adds more, and the RLVR increments sit at or above the 0.903 similarity of the two RLVR runs to each other while preserving the parent's hubs (Table 2, right): the pipeline's footprint is dominated by the SFT and DPO stages, while no RLVR increment is resolved above the single run-pair reference — a stage-resolved dose curve consistent, at production scale, with the Qwen seed-null finding (Yue et al., 2025; Venhoff et al., 2026), read as a consistency check rather than an independent replication of the null.

A signature claim can be large, statistically significant, and scale-stable — and still be an artifact of the base model: our concentration measurement had all three properties (42.7% top-1% share, $z = 20$, Gini flat across a $23\times$ parameter range) and none survived the matched base checkpoint, the one control most studies omit; a top-1%-of-dictionary denominator likewise fabricated a "scaling law" that width-independent metrics erase — both traps report a quantity at one operating point silently read as a contrast. A third trap is dictionary branding: our own detection headline survives random encoders (Table 3), so what the audit measures is activation geometry through a matched sparse projection, not trained-feature structure — we ran the random-baseline gate of Korznikov et al. (2026) on ourselves and rebranded accordingly.

These cautions instantiate broader concerns: Méloux et al. (2025) note that attribution, probing and SAEs can produce plausible explanations even for *randomly initialized* networks; ours is the training-axis instance of their warning, made constructive (matched real base, named metric, per-

mutation/width controls, cross-family replication). Against the entropy-collapse picture (Cui et al., 2025) and its behavioral counterpart (Yue et al., 2025; Wang et al., 2025), our nulls do not resolve a footprint of accuracy-only GRPO at 1.7B from its run-to-run variation — compatible with weight-space (Wei et al., 2026) and state-distribution (Nie, 2026) accounts and with Venhoff et al. (2026), though a within-band contrast does not measure the footprint's size (Sec. 3).

Given the base control, the index slots of Fig. 8 (App. M; 13 of a 15-slot panel appear in at least one graph's top-30 across a 31-layer scan; ids, not identities) admit a pretraining-crystallized reading that we have not tested beyond the matched base (Sec. 5).

Method, related work, battery and auxiliary probes.

Backbones and SAEs. We use the `Qwen3-1.7B` (Yang et al., 2025) pre-trained *base* model and its matched SFT and GRPO (Shao et al., 2024) checkpoints we trained for this study on function calling (BFCL (Patil et al., 2025), accuracy reward — not mathematical reasoning) supplying every audited Qwen checkpoint: SFT, an accuracy-only GRPO baseline, four further GRPO seeds, and one *shaped-reward* variant (*shaped-*λ0.1: accuracy reward plus an auxiliary SAE-feature shaping penalty, weight λ=0.1); the base checkpoint is the load-bearing control. *Training provenance (disclosed in full, App. D):* the SFT checkpoint is a LoRA (Hu et al., 2022) adapter merged into the base at analysis time; all GRPO runs are *full-parameter* verl updates initialized *from the base model* — SFT and GRPO are *parallel branches from base*, so base→GRPO is a clean base-vs-RL comparison (the serial-pipeline case is covered by the Tulu-3 staircase, Sec. 4.3). The six GRPO-family runs are checkpointed at heterogeneous step counts (100–300) and include one shaped-reward recipe, so the null band conflates seed, step, and recipe variance — wider than pure seed noise: conservative for *detecting* an effect, *anti*-conservative for *certifying* the RL null (Limitation L7); App. F quantifies this with the matched same-recipe, same-step comparison. The H1 co-activation graph uses the `qwen-scope-3-1.7b-base-w32k-l50` residual-stream SAEs (src layer 14 $\rightarrow$ dst 22, $d_{\text{SAE}} = 32{,}768$); the 8B arm and the scale sweep use `qwen-scope-3-8b-base-w64k-l50` and the circuit-tracer Qwen3 transcoders (Hanna et al., 2025) (App. D). Crucially the SAE (Bricken et al., 2023; Templeton et al., 2024; Gao et al., 2025) is trained on the base model only, so the *same* feature basis probes all checkpoints — exactly why the base→GRPO comparison is well-posed.

The probes (and the control they require). H_1 *(concentration + base control):* we extract a 10,000-edge co-activation graph from a pinned probe set, with edge weight $w_{ij} = \frac{1}{|X|}\sum_{x\in X} z_i^{\text{src}}(x)\, z_j^{\text{dst}}(x)$ (mean co-activation over probe states X) over base-SAE codes of last-position residual streams (per-prompt top-2,000, global top-10,000 by $|w|$; App. D). This is a co-activation (linear-probe) attribution proxy, not a gradient or causal attribution; the identical estimator is applied to every checkpoint, so all contrasts are measured alike (which does not by itself guarantee that checkpoint-dependent estimation bias cancels). We measure the top-1%-of-*active* inbound-weight share (and Gini) on *all three* checkpoints, so any RL-specific component is the base→GRPO *difference*, not the level; a degree-preserving permutation null separates "concentrated vs. chance" from "RL-caused vs. base". H_2 *and* H_3 *(auxiliary generation-side probes):* unlike H_1, these never encode the audited model's activations: each checkpoint *generates* continuations of a probe set, a frozen external reader (Pythia-70m with the public `pythia-70m-deduped-res-sm` residual SAE, the v0 pilot pair) encodes the generated text, and a 32-feature panel preselected by a function-word-outcome screen — features ranked by association with the next token falling in a closed set of function words, over 2,048 rollouts at sequence length 64 (positive rate 0.092); the selection receipt, including all 32 indices and the screen's definition, is committed at `notes/panel_v0_pythia70m.json` and subsequently evaluated on the v0 pilot graph in the *reader's* SAE space is measured (App. D, M). H_2: per-feature mean activation (16 probe prompts); H_3: per-feature entropy of per-sequence mean activation over 24 prompts $\times 4$ continuations, testing the $|d| \geq 0.5$ collapse gate (a $|d| = 0.5$ threshold we pre-registered ourselves, following Cohen's conventional medium effect; Yue et al. (2025); Wang et al. (2025) report behavioral results and state no feature-level gate, so this bar is ours to defend, not theirs) — both probe post-training's effect on the *generated prompt-plus-continuation text* through a fixed SAE lens.

The control generalizes across recipes/seeds and families. The null is not specific to `Qwen3-1.7B` and the `qwen-scope` SAE; we replicate it along two axes (Table 5, Fig. 3; App. H).

Within Qwen, we measure the same base→post deltas for five independent post-training runs sharing the base-trained SAE: SFT, accuracy-only GRPO, two further GRPO seeds, and the shaped-reward recipe (shaped-λ0.1, App. D); all five give $|\Delta\text{Gini}| \leq 0.005$ and top-32 hub Jaccard 0.94–1.00 (three at exactly 1.00) — the razor-thin null is not a single-seed accident. (The displacement null of Sec. 4.2 adds seeds s1/s3; full run roster with steps in App. D.) *Across families*, we run the identical base-vs-post control on `Gemma-2-2B` (Gemma Team et al., 2024) with the `gemma-scope-2b-pt-res` SAE (Lieberum et al., 2024) — a different family, SAE architecture (JumpReLU), and dictionary width (16k). The zero-RL `Gemma-2-2B` base already carries essentially the same concentration as the Qwen base (Gini 0.708 vs. 0.728, top-1%-active 0.347 vs. 0.427): the base-set concentration is not specific to one backbone or SAE.

One honest nuance. The Gemma post-checkpoint we can access is the production `gemma-2-2b-it` instruct model (heavy SFT+RLHF+distillation), *not* a matched light GRPO run; its base→post shift is genuinely larger (ΔGini $= +0.035$, hub Jaccard 0.73). So a post-trained checkpoint can move the level (the heavy instruct `-it` checkpoint does, light GRPO does not), by an order of magnitude less than the single-checkpoint level; with one cross-family pair we do not attribute the difference to post-training intensity; the *level* is base-set in both families, so a single-checkpoint level is never safe to read as an RL effect. A further caveat: `gemma-2-2b-it` is a chat model probed on raw (untemplated) pile text, so part of its larger shift may be chat-template distribution mismatch rather than post-training per se; a chat-templated re-probe is the clean follow-up.

Concurrent base-vs-RL studies (the claims our control tests). Attribution graphs (Ameisen et al., 2025; Lindsey et al., 2025; Heimersheim & Nanda, 2024; Vig et al., 2020) and sparse feature circuits (Marks et al., 2025) extract the feature panels we audit; following Munigety (2026), we frame H_1 as a reweighting finding about attribution mass, not a direct causal claim, and treat H_2 activation deltas as a polarity probe (App. G). Two closely related works diff a model against a post-trained descendant and report internal change — Zhang et al. (2025a) (before vs. after RL fine-tuning: RL increases activation *intensity/diversity*, online RL > DPO) and Hu et al. (2026) (base vs. reasoning-distilled: adapter features for hesitation tokens are necessary and sufficient) — but neither reports a matched-base concentration-*level* control; our matched-base control locates the concentration *level* in the base, and the only larger *residual* we observe is on a confounded cross-family `gemma-2-2b-it` checkpoint (heavy SFT+RLHF+distillation), which we do not attribute to intensity or to an online-RL>DPO ordering (Table 5 carries no matched DPO comparator).

Entropy-collapse and the narrowing-feature picture. Yue et al. (2025) argue, from pass@k curves, that RLVR does not expand the base model's reasoning capacity beyond what sampling already reaches; Wang et al. (2025) show a single training example can recover most of the RLVR gain; Lin et al. (2024) show base models can be aligned in-context without weight updates. *These are behavioral results, and none of them measures SAE features* — but together they make a feature-level prediction that the internals literature has since asserted directly: if RL only reweights what the base already computes, the features it uses should narrow. Cui et al. (2025) supply the optimization-side version, policy-entropy collapse. That extrapolated prediction is what our probes test. Our cross-tokenizer entropy probe finds no collapse (committed cross-tokenizer run $\Delta H = -0.004$ nats, $d = -0.005$; App. M); since the entropy drop is on-policy-specific (G. & Lindsey, 2026), our null may be regime-dependent — a matched on-policy/off-policy H3 control is pre-registered; the entropy–gradient readout of Yang et al. (2026) is future work.

H2/H3 auxiliary nulls (App. M). Two panel-internal probes read the real `Qwen3-1.7B` checkpoints' *generated text* through a fixed external SAE reader (Pythia-70m with the `pythia-70m-deduped-res-sm` SAE; the 32-feature panel and its indices live in the reader's space) and agree with the headline: activation deltas are polarity-preserving (weighted cosine 0.9998), and the $|d|$=0.5 collapse gate (a $|d| = 0.5$ threshold we pre-registered ourselves, following Cohen's conventional medium effect; Yue et al. (2025); Wang et al. (2025) report behavioral results and state no feature-level gate, so this bar is ours to defend, not theirs) is never approached: across the committed accuracy-only GRPO runs the per-feature entropy delta is small and sign-unstable (cross-tokenizer run $d = -0.005$; probe-budget reruns of the same baseline checkpoint span $d = -0.19$ to $+0.32$), and a reward-shaped regime reaches only $d = -0.29$ — all far below the gate. Being output-side probes on a pilot panel, they are corroboration, not internal-activation nulls; full tables, provenance, and two *withdrawn* earlier-draft claims in App. M.

F.1 BASE CONTROL FOR THE ACTIVATION-SIDE "NARROWING" METRICS

The base control above concerns co-activation-graph *concentration*, but the RL-internals literature reads RL as "narrowing" through an activation-side family of single-checkpoint metrics (features fire more selectively, usage sparsifies, entropy drops), our feature-level extrapolation of behavioral and policy-entropy results (Yue et al., 2025; Wang et al., 2025; Cui et al., 2025). We run the same base-vs-post control on a battery computed from SAE encodings (n=500, ∼32k tokens, dst-layer SAE): firing-frequency Gini, mean-activation Gini, and usage entropy (Table 8 and Fig. 6b, App. H). *A metric we must exclude, and why.* Mean L_0 — the most commonly quoted "sparsification" statistic — is *uninformative by construction* on our headline Qwen dictionary: `qwen-scope-3-1.7b-base-w32k-l50` is a fixed-k TopK SAE (Gao et al., 2025) with k=50, so its L_0 is a property of the dictionary, not of the model being probed. It reads exactly 50.0 on all 20 per-checkpoint Qwen receipts we computed — base, SFT and five GRPO-family runs across three probe distributions (pile, GSM8K, BFCL train and validation) — and its seed band is literally $[0, 0]$: it *cannot* move, so a null on it is not evidence of anything. We therefore drop mean L_0 from the Qwen battery and report it only where the SAE's sparsity is data-dependent and it can genuinely vary — Gemma's JumpReLU dictionary and `Llama-Scope` (Sec. 4.3), where it does vary. On the metrics that can move, the result is, in both halves, *negative for narrowing*: (i) on the light-RL trio every metric is base-dominated (firing-freq Gini 0.683 throughout, $|\text{rel } \Delta| < 1\%$); (ii) on the heavy-instruct Gemma pair the two metrics that move most shift *opposite* to the prediction (JumpReLU mean L_0 *rises* 184.8 → 198.6, firing-freq Gini *falls* 0.494 → 0.486; mean-activation Gini and usage entropy move ≤ 0.002). "RL narrows the SAE basis" is not observed in either regime.

G EXTENDED RELATED WORK

The sharpest apparent conflict with our RL null. Zhang et al. (2025a) is the sharpest apparent conflict: in the same method family they find online RL raises activation intensity and diversity, while *also* reporting DPO "deviate[s] from these trends"; we do not read that as confirming our dose-dependent DPO reading. Their readout is activation summaries, not a graph contrast against the algorithm's own band, and their RL is production-scale to our light GRPO; we report their positive as live for their regime and scope ours to what we null-calibrate (L1, L7).

Matched configuration across the SFT and DPO arms. Both arms are LoRA runs, so the adapter is a candidate carrier: Shuttleworth et al. (2025) find LoRA adds "intruder" directions absent from full fine-tuning, and K (2026) find LoRA dictionaries align weakly with the pretrained SAE basis our graphs use. Matched across the arms: base model, LoRA rank 16/α 32, target modules (`q,k,v,o_proj`), step count (29), effective batch (32), graph protocol, SAE and scoring code. Not matched, disclosed rather than buried: lr (10×, already L8) and `lora_dropout` (SFT 0.05 vs. DPO 0.0, inherited from the upstream scripts); the lr difference plausibly pushes *toward* the observed gap (a smaller lr mechanically reduces DPO's displacement), so the step-matched contrast is between the methods' standard recipes, not an lr-matched isolation of the objective; neither changes adapter rank or placement, and K (2026) find the divergence from the pretrained SAE basis relatively stable across LoRA ranks 4–32, so rank is an unlikely carrier of the gap. The claim is about two objectives at matched adapter geometry and dose, and licenses nothing about full fine-tuning.

Attribution-graph interpretability. Attribution graphs — canonically the circuit-tracing line of Ameisen et al. (2025); Lindsey et al. (2025), with activation-patching and causal-mediation precursors (Heimersheim & Nanda, 2024; Vig et al., 2020) and the SAE-feature-level circuits of Marks et al. (2025) — are widely used to extract feature panels; we use 10,000 edges and the top-32 destination features. Munigety (2026) show on GPT-2 IOI that the most *selective* SAE features are not the most *causal*, so we frame H_1 (our concentration measure; Sec. 3) as a reweighting / concentration finding about attribution mass, not a direct causal claim, and treat H_2 activation deltas as a polarity probe.

The null-calibration gap in the RL-internals literature (full positioning). Shi et al. (2026) is the closest prior art: training SFT and RL from the same base on identical data (a strong design), they find RL induces *restrained* feature changes that preserve base structure while SFT introduces many new features — directionally consonant with our SFT-positive/RL-null result, but without

asking whether the residual RL changes exceed seed noise. Rojas Nunez et al. (2026) reach the same ordering from a different direction entirely: their *differential circuit vulnerability* is a head-level degradation measure, applied to `Qwen2.5-3B-Instruct` on scientific QA, and it finds SFT adapting faster while disrupting circuits substantially more than Dr.GRPO, which retains more of the base circuit. Three scope differences are worth naming, none of which we read as a conflict: their RL arm is *serial* (Dr.GRPO refining an SFT policy) where our Qwen SFT and GRPO are parallel branches from base; their comparison contains no preference-optimization arm, so it says nothing about the SFT/DPO ordering that is our dose-matched result; and the unit is attention heads rather than sparse features. That two groups converge on the direction from designs sharing neither backbone, unit, nor comparator is the strongest kind of agreement available — and it is agreement about a direction, which is a different thing from a width. Supplying the width is what the seed-null protocol here does, and it is why the SFT/DPO statement we lead with is a disjointness over four seeded runs per arm rather than a ratio between two runs. Shportko et al. (2026) localize RL-induced tool use to a compact crosscoder feature set on a matched base/RL pair, yet themselves flag the missing maximally-elicited-base control and decline the strong installation claim. Yan et al. (2026) give architecture controls (Llama, OLMo-2 negatives for a Qwen-specific shortcut mechanism); Wang et al. (2026a) show weights and output log-probabilities drift linearly under RLVR across configurations (consistency across runs, not a null for structure); Huang et al. (2026) find sparse directional token-level updates; Sinha et al. (2026) find six alignment algorithms leave different probe/SAE-anchor geometric traces across three families. Tam (2026)'s shallow-alignment finding (alignment training deactivates policy features while leaving latent partisan structure intact) is rhetorically adjacent, and our seed-null converts that style of claim into a calibrated one. On the elicitation side, Venhoff et al. (2026); Zhao et al. (2025); Zhang et al. (2025b) show reasoning behaviors can be turned on *without weight updates*, and Wu et al. (2025) show pretrained SAE-rule structure *predicts* post-RLVR performance — behavioral and predictive evidence to which our audit adds the missing measurement-side complement: a null-calibrated post-training diff in sparse feature space, on one full public staircase. Finally, the crosscoder-diffing line (Minder et al., 2025) shows naive L1 crosscoders *fabricate* chat-specific latents (an artifact class our base-trained-SAE design avoids by construction), reinforcing that model-diffing claims need explicit nulls.

Tokenizer alignment, weight-space, and state-distribution accounts. Pre/post checkpoints can tokenize identical strings differently (e.g. added chat-template tokens), activating *different* SAE features; we adopt explicit cross-tokenizer alignment. Our non-detection is compatible with weight- and state-space accounts: Wei et al. (2026) find a rank-1 approximation of the RLVR weight delta captures most of its gain; Nie (2026) reframe post-training as state-distribution reshaping; Li et al. (2026) find that a random 20% of tokens preserves much of the full-token RL update and that low-attention-entropy "anchor" tokens carry its stable backbone — the token-space counterpart of our feature-space concentration. Krishnamurthy et al. (2026) decompose RL into *strategy selection* (from an existing repertoire) vs. *improvement* (new patterns from hard data); our base-present concentration and absent entropy collapse are consistent with a *selection*-dominant regime, on which reading novel-circuit signal should grow as RL data is made harder. At the behavior level Kirk et al. (2024) established the RLHF-vs-SFT contrast on generalization and output diversity, Li et al. (2025) show explicit chain-of-thought reasoning can *degrade* instruction-following accuracy, and Jiang et al. (2026) show DPO raises the model's internal value estimate, along a pre-existing expected-success axis, for rewarded behaviors — a behavior-space instance of the "post-training reweights a pre-existing direction" picture our H1 measures; the recurring theme is that post-training moves behavior substantially even where representational change is small (our own data do not test this: the GRPO non-detection does not measure the footprint, and the behavioral and internal arms are on different state distributions; L5). The reward is itself a design axis separable from the RL update rule; our single shaped-$\lambda 0.1$ run moves H3 slightly more ($d = -0.29$) than the accuracy-only baseline, within the spread of our probe-budget reruns (App. M). The H1 hubs may also be *parent* nodes in a sparse feature hierarchy; hierarchy-aware accounting is left for future work.

H Additional H1 material

The dictionary-width artifact in full. Independently of the base control, an earlier draft reported a "$51.1\% \rightarrow 84.0\%$" jump from 1.7B to 8B as evidence that concentration intensifies with scale. Re-deriving both numbers from the committed graphs shows *three* confounds stacked, not one. (i)

Table 4: **Headline control: concentration is base-set across training stages** (`Qwen3-1.7B`, `qwen-scope` src 14 → dst 22, n=500, base-trained SAE). The zero-RL base already shows the full concentration; RL-attributable deltas are negligible and 31–32 of the 32 top hubs are shared (Jaccard 0.94–1.00). An uncontrolled single-checkpoint analysis would have read the GRPO column alone as an RL signature.

Checkpoint	top-1% (active)	Gini	# active dst features
Base (zero RL)	0.427	0.728	2226
SFT	0.416	0.723	2229
GRPO	0.429	0.730	2225
Δ(GRPO−base)	+0.002	+0.002	—
Top-32 hub Jaccard: base∩GRPO 0.94, SFT∩GRPO 1.00, base∩SFT 0.94			

Table 5: **The base control replicates across 5 Qwen recipes/seeds and a second family.** Width-robust deltas (base→post) of the same co-activation-graph concentration. *Within Qwen* (`qwen-scope`, 32k, n=500): all five post-training runs give $|\Delta\text{Gini}| \leq 0.005$ and hub Jaccard 0.94–1.00. *Cross-family* (`Gemma-2-2B` + `gemma-scope`, JumpReLU, 16k, n=500): the zero-RL base already carries the concentration (Gini 0.708); the larger `-it` shift reflects heavy production instruct-tuning, not light GRPO; with one heavy checkpoint and a possible chat-template confound (App. F) we do not attribute it to intensity.

Family / SAE	Checkpoint	Post-training	ΔGini	Δtop-1%	hub Jaccard
Qwen3-1.7B / qwen-scope	base	— (Gini 0.728)	—	—	—
	SFT	SFT (LoRA)	−0.005	−0.011	0.94
	GRPO	accuracy-only GRPO	+0.002	+0.002	0.94
	GRPO-s4	GRPO (seed)	0.000	+0.001	1.00
	GRPO-s5	GRPO (seed)	+0.001	+0.002	1.00
	shaped-λ0.1	shaped-reward GRPO	0.000	0.000	1.00
Gemma-2-2B / gemma-scope	base	— (Gini 0.708)	—	—	—
	`-it`	heavy SFT+RLHF+distill	+0.035	+0.062	0.73

Provenance: the 51.1% endpoint is not a 1.7B number at all — it is the top-1%-of-active share of the v0 `Pythia-70m` pilot graph (0.5111, `notes/attribution_graph_v0.json`), a different backbone and SAE family. (ii) *Denominator*: the 84.0% endpoint is top-1%-of-*dictionary* (0.8395 on the 8B graph), so the two endpoints did not share a denominator either. (iii) *Width*: with the denominator held fixed at top-1%-of-d_{SAE}, the honest 1.7B→8B pair is 0.7110 → 0.8395, and the remaining rise is mechanical — 1% of the 2×-wider 8B dictionary is 655 features against 328. Under any width-independent statistic the effect reverses or vanishes. We record the corrected numbers rather than quietly deleting the claim: the failure mode being illustrated is exactly a number whose provenance was never re-checked. We measure concentration at matched relative depth (src ≈ $0.5L$, dst ≈ $0.79L$, n=500, 10,000 edges) across two SAE families and 0.6–14B (Table 7): in the `qwen-scope` residual family the Gini of inbound weight is 0.728 at 1.7B and 0.719 at 8B; in the circuit-tracer transcoder family it is 0.688/0.654/0.645 at 0.6/1.7/14B. Both families are flat-to-slightly-decreasing across a 23× parameter range—this sparse sweep shows no monotonic increase of concentration with scale.

The full seven-metric seed-null battery. Applying the differencing framework to the informative part of the activation-side battery (five GRPO-family runs re-measured per checkpoint, pile probe set): for all *seven* metrics that can vary — dead fraction, firing-frequency Gini and top-1% share, mean-activation Gini, normalized usage entropy, and two activation-*intensity* levels (total activation mass per token; mean activation per firing event) — the base→GRPO delta lies within the GRPO-seed pairwise band (e.g. activation mass per token: $|\Delta_{\text{base}\to\text{GRPO}}| = 0.31$ vs. seed band $[0, 3.09]$; mean act per firing: 0.006 vs. $[0, 0.062]$). The receipt `notes/featstats_seednull.json` carries an eighth row, mean L_0, at 50.0 with a $[0, 0]$ band; we exclude it from the count because the TopK dictionary pins it (Sec. 4.1) — it is not a null result, it is a constant. Three of the seven retained metrics are themselves close to the precision floor (Sec. 3, "a statistic must beat its own reporting precision"): firing-freq top-1% and mean-activation Gini have $|\Delta_{\text{base}\to\text{GRPO}}|$ equal to the seed band

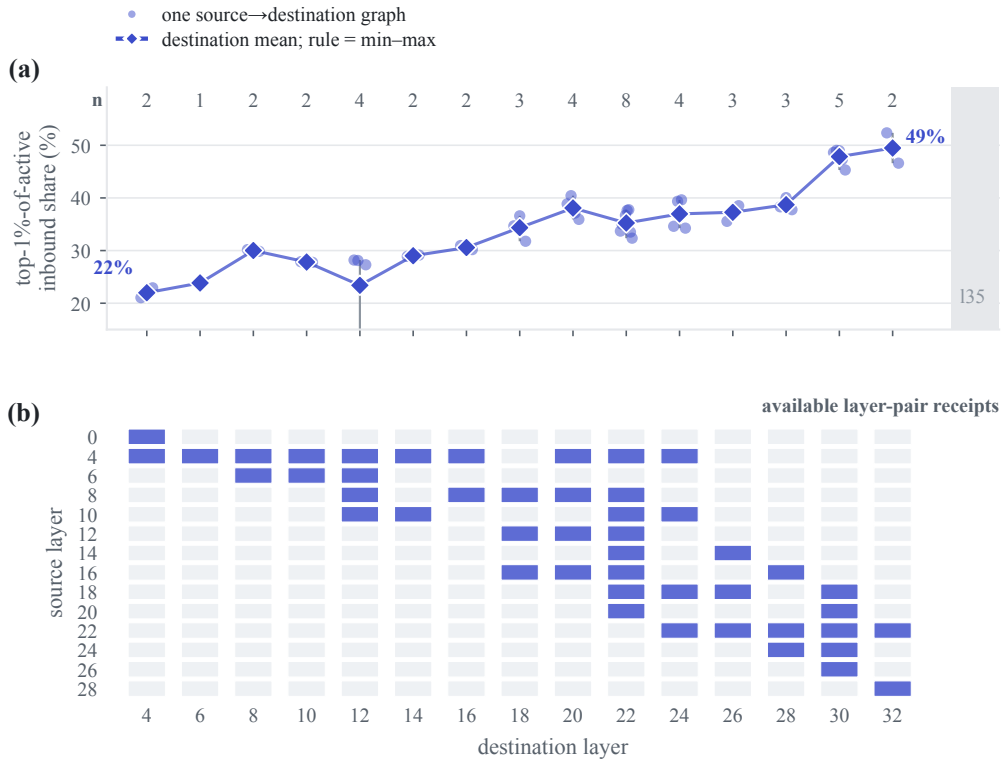


Figure 5: **Available layer-pair coverage and concentration.** Every point is one source-to-destination co-activation graph; diamonds are destination means and rules the observed min–max range. Numbers above the plot count available source layers. The lower panel shows the unbalanced source/destination coverage. The rise from about 22% to 49% is descriptive of this available scan, not an isolated destination-depth effect. Terminal layer 35 is excluded as unembedding-dominated. Source: the 47 committed Qwen3-8B layer-pair receipts (`notes/attribution_graph_v1_qwen3-8b_l*to*.json`); generator `code/analysis/fig7_concentration_vs_depth.py`.

Table 6: **H1 edge-weight percentile distribution (left) and the six destination features with the largest total inbound weight (right)** (`Qwen3-1.7B` + `qwen-scope-3-1.7b`, src 14 → dst 22). The two halves are independent summaries of the same 10,000-edge graph; rows do not correspond across the divider.

Percentile	Edge weight	Feature ID	Total inbound w	In-degree
p_{25}	14.3	30329	71560	2061
median	17.2	26766	9453	241
p_{75}	23.9	10420	3212	88
p_{90}	36.4	25789	2537	80
p_{99}	112.2	11799	1679	48
max	3597	23554	1109	36

maximum at 0.0002, one unit of the reported precision, so for those the honest statement is "no movement resolvable at our precision," not "inside a measured band." The four metrics with bands comfortably above their reporting step (dead fraction, firing-freq Gini, and the two intensity levels) carry the claim. We do not claim Zhang et al. (2025a)'s intensity result is wrong at their heavier operating point — our Gemma arm shows heavy post-training does move activation statistics — but the contrast shows their headline does not generalize downward to light GRPO.

Table 7: **H1 concentration is scale-stable.** Top-1% (of active features) inbound-weight share and Gini at matched relative depth (n=500, 10,000 edges), across two SAE families and 0.6–14B. Both families are flat-to-slightly-decreasing with scale; the dictionary-relative "top-1%-of-d_{SAE}" (last column) rises only because wider dictionaries place more features in a fixed 1%. The `qwen-scope` (Deng et al., 2026) rows are the committed receipt `gini_width_control_qwen_1p7b_vs_8b.json`; the transcoder rows (circuit-tracer Qwen3 transcoders (Hanna et al., 2025)) come from the sweep job scripts in the supplement (no committed per-run summary JSON).

SAE family	Model	top-1% (active)	Gini	top-1% (dict)
qwen-scope (resid)	Qwen3-1.7B	42.7%	0.728	71.1%
qwen-scope (resid)	Qwen3-8B	35.2%	0.719	84.0%
transcoder (circuit-tracer)	Qwen3-0.6B	35.9%	0.688	—
transcoder (circuit-tracer)	Qwen3-1.7B	31.1%	0.654	—
transcoder (circuit-tracer)	Qwen3-14B	32.6%	0.645	—

Table 8: **Activation-side "narrowing" metrics are base-dominated (light RL) or move *against* narrowing (heavy instruct).** Base→post deltas of four common single-checkpoint SAE metrics (n=500, dst-layer SAE). *Qwen3-1.7B* (light GRPO, `qwen-scope` l22): all metrics base-dominated, $|\text{rel }\Delta| < 1\%$. *Gemma-2-2B* (heavy SFT+RLHF+distill, `gemma-scope` l20): mean L_0 *rises* (less sparse) and firing-freq Gini *falls* (less concentrated) — the opposite of the predicted RL "narrowing". Narrowing arrow shows the direction RL would move each metric under the entropy-collapse/narrowing hypothesis. *The mean-L_0 row is informative only in the Gemma columns*: `qwen-scope` is a fixed-k=50 TopK dictionary, so its L_0 is pinned by construction and its Qwen entries (greyed) are a property of the SAE, not a measurement of the model; Gemma's JumpReLU L_0 is data-dependent and does move.

	narrowing	Qwen3-1.7B / qwen-scope l22			Gemma-2-2B / gemma-scope l20	
Metric	predicts	base	ΔSFT	ΔGRPO	base	Δ`-it`
mean L_0	↓	50.0†	+0.0†	+0.0†	184.8	+13.8
firing-freq Gini	↑	0.683	−0.000	+0.000	0.494	−0.007
mean-act Gini	↑	0.733	−0.001	+0.000	0.643	+0.002
usage entropy (norm)	↓	0.895	+0.000	−0.000	0.950	−0.000

† structurally pinned: TopK dictionary with k=50; excluded from all Qwen metric counts.

GSM8K domain robustness in full. A natural objection is that RL would narrow only the features *relevant to its task*, so a single probe distribution could miss a localized narrowing. We therefore re-run the entire battery on the clean Qwen base/SFT/GRPO trio with a second, structurally different probe set — GSM8K question+solution chains (n=500, ∼32k tokens). *An honest scope note:* the RL task here is BFCL function calling (Sec. 3), so GSM8K is *not* the RL training distribution — it is a second off-training-distribution probe that engages a structured, step-by-step-reasoning region of the feature basis, and the *on*-distribution (function-calling) probe is the on-task BFCL audit of App. I. The GSM8K distribution is indeed far more concentrated than pile (base firing-frequency Gini 0.840 vs. 0.683; mean-activation Gini 0.883 vs. 0.733; dead fraction 0.36 vs. 0.15), confirming it genuinely engages a different region of the basis. Yet the base→post deltas remain nil: firing-frequency Gini is 0.840 throughout ($\Delta \leq 0.0004$), and every base→GRPO metric has $|\text{rel }\Delta| < 1\%$ (the four headline narrowing metrics are $< 1\%$ at every stage; SFT's dead fraction rises $+4.7\%$ and its firing top-1% falls -1.5%, the largest GSM8K moves). The narrowing-metric null is therefore not an artifact of one probe distribution: it holds across two probe domains with very different feature-usage profiles; the probe on the RL task distribution itself is App. I.

I On-task (BFCL) arm and its minimum detectable effect

Protocol. Same estimator, checkpoints, and base-trained SAE as the headline audit (Sec. 3); only the probe states change: the RL task's own function-calling prompts (the verl BFCL training parquet, n=500 prompts for graphs and battery; the n=100 held-out validation split for the val arm), rendered

Table 9: **Co-activation displacement is within run noise for five of six RL contrasts (s1 falls $0.2\times$ the null range below the band, vs SFT's $4.0\times$); SFT clears the null at $z = -15.9$.** Pairwise graph similarity (`Qwen3-1.7B`, l14→22, 10,000 edges) vs. the 15-pair GRPO-family null. Negative z = more displaced than seed noise; only SFT separates by a large margin, while s1 falls marginally below the band at step 200 ($z = -2.0$); at matched step 300 it is inside the 15-pair band ($z = -0.98$) and 0.0006 below the three-pair same-step band.

Pair	edge Jaccard	z vs. seed null	rank[§]
base→SFT (LoRA)	**0.937**	**−15.9**	$\leq 1/16$
base→GRPO	0.962	−0.5	0.38
base→GRPO s1/s3/s4/s5	0.959–0.964	−2.0 to +0.8	0.06–0.81
base→shaped-λ0.1	0.964	+0.8	0.81
base→SFT vs. *SFT's own* null	**0.937**	**−19.9**	$\leq 1/7$
base→SFT s0/s1/s2/s3 (own null)	0.937–0.941	−20.0 to −17.0	$\leq 1/7$

Seed-noise null: 6 GRPO-family runs, 15 pairwise; edge Jaccard 0.9626 ± 0.0016, cosine 0.999.
SFT seed null: 4 LoRA-SFT runs, 6 pairwise; edge Jaccard 0.9644 ± 0.0014, band $[0.9619, 0.9656]$.
[§]rank quantile among the $m+1$ values; a tail probability only if the contrast is exchangeable with the seed pairs, which run exchangeability does not imply (Proposition 1: attainable bound $1/k$).



Figure 6: **A run-variation reference and the diagnostic battery.** Same-family null pairs (n=15) are distinguished from the base-to-SFT contrast and the base-to-RL contrasts, including a separately marked shaped-reward arm. The battery retains each checkpoint's own base and explicitly marks TopK-fixed and unavailable values. The SFT shift exceeds this empirical range; these diagnostics do not establish the absence of a smaller RL effect. Receipts: `notes/circuit_rewiring_qwen_6seed.json` and the `featstats_*/tulu_featstats_*` JSONs; generator `code/analysis/fig_seednull_battery.py`.

through the checkpoint tokenizer's chat template with the generation prompt appended, sequence length 1024 keeping the *last* tokens — so the graph's last-position state is the function-call decision point, the exact input state verl used for rollouts. The base-trained SAE remains a usable lens on these states (FVE 0.81–0.84). Co-activation graphs (l14→22, 10,000 edges) are built for base, SFT, and the same six GRPO-family runs as the pile arm, giving the same 15 seed-null pairs.

Outcomes, reported descriptively (Tables 10, 11). *Displacement:* base→GRPO edge Jaccard 0.746, $z = -0.23$, inside the seed band $[0.667, 0.819]$; the other five base→RL contrasts fall at $z \in [-0.87, +1.83]$, none below the band's more-displaced lower edge (base→s4, 0.829, sits *above* the band's upper edge at $z = +1.83$; caveat iii). *Levels:* against the base's Gini 0.755 (top-1%-active 0.217), all seven post checkpoints have $|\Delta\text{Gini}| \leq 0.026$ with top-32 hub Jaccard 0.829–0.882. *Battery:* the base→GRPO delta is within the five-run seed-pair band on all eight informative metrics (the ninth, mean L_0, is TopK-pinned at 50.0 with a $[0, 0]$ band and is excluded, Sec. 4.1); the validation-split trio is flat (firing-freq Gini $0.8532/0.8529/0.8541$ at base/SFT/GRPO over 41,612 identical tokens; FVE 0.81–0.84).

Why the arm is uninformative, quantified. (i) The on-task seed-noise band is ∼25× wider than pile's (SD 0.0403 vs. 0.0016; range 0.152 vs. 0.006): task states are intrinsically higher-variance across runs, so the 2σ minimum detectable effect is $\Delta\text{Jaccard} = 0.0806$ on-task (10.7% of the statistic) vs. 0.0032 on pile. SFT's actual pile-arm effect is 0.0254 (2.6% of that statistic): the on-task arm can only detect effects ∼3× larger in absolute Jaccard (∼4× relative to each arm's statistic) than the one the audit's positive control produces — it is structurally incapable of adjudicating an effect of the size at issue. (ii) Consistently, the SFT positive control does *not* separate on-task: base→SFT 0.712, $z = -1.06$ (pile: $z = -15.9$, four times outside the null range). This is mechanistically consistent with the pile localization (App. J) — SFT's detected displacement prunes boilerplate/function-word edges into the generic-syntax hub, wiring that BFCL's structured task prompts barely exercise — but it means the on-task arm carries no internal demonstration that it could detect a real signature. (iii) The arm's null band is itself contaminated: base→s4 lands at $z = +1.83$, *above* the band's upper edge — a base→RL contrast more base-like than the seed pairs are to each other, reflecting the step/recipe mixture of Limitation L7 — and the hub-Jaccard statistic takes only three distinct values across the 15 null pairs, too quantized to score. We therefore class the on-task arm as uninformative at the relevant effect size and draw no conclusion from it in either direction; the RL-null claim rests on the pile arm.

Table 10: **On-task (BFCL) seed-noise-null displacement vs. the pile arm** (`Qwen3-1.7B`, l14→22, 10,000 edges, edge Jaccard). Reported for completeness, not as a null: the on-task arm's 2σ MDE (0.0806) is ∼3× SFT's actual pile-arm effect (0.0254; ∼4× in relative terms), and SFT does not separate on it ($z = -1.06$), so no conclusion is drawn from this table (caveats i–iii).

	BFCL task states	pile
seed null, 15 pairs (mean ± SD)	0.7551 ± 0.0403	0.9626 ± 0.0016
seed band $[\min, \max]$	$[0.667, 0.819]$	$[0.961, 0.966]$
base→GRPO (z)	0.746 (-0.23)	0.962 (-0.5)
base→s1/s3/s4/s5/shaped (z range)	-0.87 to $+1.83$	-2.0 to $+0.8$
base→SFT (z)	0.712 (-1.06)	**0.937** (**-15.9**)

Table 11: **On-task (BFCL) levels and battery.** Left: per-checkpoint graph levels vs. base (Gini 0.755, top-1%-active 0.217). Right: base→GRPO battery deltas vs. the five-run GRPO-family seed-pair band (train split, n=500); all eight informative metrics within seed noise. The mean-L_0 row (greyed) is TopK-pinned at 50.0 with a $[0, 0]$ band and is excluded from the count.

Checkpoint	ΔGini	hub Jacc	Battery metric	$\lvert\Delta_{\text{base}\to\text{GRPO}}\rvert$	seed max
SFT	-0.026	0.829	mean L_0 (pinned)	0.0	0.0
GRPO	$+0.007$	0.882	dead fraction	0.0046	0.0070
s1	-0.002	0.882	firing-freq Gini	0.0003	0.0004
s3	$+0.006$	0.882	firing-freq top-1%	0.0040	0.0051
s4	$+0.013$	0.882	mean-act Gini	0.0010	0.0012
s5	$+0.017$	0.829	usage entropy (norm)	0.0011	0.0012
shaped-λ0.1	$+0.026$	0.829	act mass / token	16.9	21.4
			mean act when active	0.34	0.43
			FVU	0.0039	0.0069

Receipts. `notes/circuit_rewiring_bfcl.json` (displacement + seed null), `notes/basediff_bfcl_levels.json` (lev-

els), `notes/featstats_seednull_bfcl.json` (battery), computed from `notes/bfcl_graph_*_l14to22.json` (8 checkpoints) and `notes/bfcl_featstats_*_l22.json` / `*_l22_val.json`.

J LOCALIZATION OF THE SFT SIGNATURE

Pruned inbound wiring at the dominant hub. We define the GRPO-family *consensus* edge set (edges present in all 6 GRPO-family graphs; 9,584 of 10,000) and isolate edges that base and every GRPO run retain but SFT drops from its top-10k graph (138 edges, 0.78% of base edge weight). The drops are strikingly localized: 86 of 138 (63% of dropped weight) are inbound edges to a single destination feature — #30329, the rank-1 concentration hub itself (Table 6) — and the top-8 destination features hold 71.7% of all dropped weight. SFT-*added* edges (in SFT, absent from base and all GRPO runs; 124 edges) are by contrast diffuse (top-8 share 24%). So the SFT signature is a *localized pruning of inbound wiring at the dominant pretraining hub* ($\approx$ 4% of the hub's in-degree pushed below the graph cutoff) — consistent with SFT being the only checkpoint whose concentration moves *down* (top-1% 0.427 $\rightarrow$ 0.416, Table 4) — while GRPO leaves the hub's wiring intact. (Caveat: "dropped" means fell below the top-10k edge cutoff, i.e. a real relative weakening, not necessarily to zero.)

What the displaced features are (max-activating contexts over the pile probe set, base-model encodings). The pruned hub #30329 is a generic *numeric / whitespace / formatting* feature (top activations on digits and leading whitespace before numbers — "`estimated` 257", "`93560`", "`ORS` 192" — and URL/markup tokens; fires on 62% of tokens, mean activation 183, by far the largest in the graph). The two source features carrying the most dropped weight are likewise high-frequency and generic — each loses 16 edges to *other* destinations, while its own edge into #30329 survives in the SFT graph at reduced weight (3076 $\rightarrow$ 2858, 3597 $\rightarrow$ 3388): #20227 fires on boilerplate/function tokens (" by", ":", " function", cookie-notice and code-scaffold text; 55% firing rate) and #16308 is near-ubiquitous (98% firing rate). So the concrete picture is: *SFT weakens the co-activation edges by which ubiquitous boilerplate/function-word features feed the base model's dominant generic-syntax hub* — a plausible footprint of instruction-tuning learning to down-weight boilerplate — and GRPO leaves exactly these edges in place. This turns the audit from a yes/no detector into a lens: when a post-training stage *does* leave a signature, the same machinery names the features it acts on.

Does the displacement carry computation? The registered causal test of the displaced sources, its pilot and the full sweep are in App. K.

A causal probe keeps this claim honest — and bounds it. We tested whether the co-activation edge corresponds to a causal pathway by ablating the source features' decoded contributions at the source layer (subtracting $z_f W_{\text{dec}[f]}$ from the residual, reconstruction error left intact) and reading the hub's activation downstream (n=200 prompts). The interaction is real and feature-specific — ablation shifts the hub 3.8–15.8$\times$ more than a count-matched random-feature control — but it is *suppressive* (removing the sources *raises* the hub by 4.9–5.7%; the co-activation edge weight, positive by construction on non-negative codes, is sign-blind), and, decisively, its causal strength is *not weaker in SFT* (hub rise +5.7%) than in base (+4.9%). So the SFT edge-drop that the audit detects is a change in *co-activation statistics*, not (by this probe) a pruned causal coupling. We therefore keep the mechanism claim strictly correlational, and flag the general lesson: co-activation-graph edges can move without the underlying causal interaction moving, one more reason signature claims need causal probes before mechanistic interpretation. (Caveats: the random control is count- but not magnitude-matched, so the specificity ratios are upper bounds; output-level KL under ablation is small, 0.010–0.013.)

K THE REGISTERED FUNCTIONAL-TRANSFER SWEEP

What was registered. Bayat Makou et al. (2026) show that structurally distinct circuits can implement the same computation, so structural displacement, at any number of standard deviations, does not identify a changed mechanism. Before running anything we registered the test they prescribe

(PREREGISTRATION_functional_transfer.md: faithfulness gate first, branches F1–F4) and, after a power pre-flight showed that the base and SFT top-k graphs share $\sim$94% of their source features, amended it to the *differential* sources. D_{SFT} (D_{base}) is the set of source features of edges in the SFT (base) top-k graph and not the other, minus sources in both. For model M and set S, $\Delta(M, S)$ is the mean final-token $\mathrm{KL}(p_M \,\|\, p_M^S)$ when, at layer 14, the SAE contributions $z_f W_{\mathrm{dec}}[f]$ of $f \in S$ are replaced by those of a resampled prompt (reconstruction error untouched). The gate asks each own-condition set to beat a size- and firing-frequency-matched random set R in its own model; the decisive statistic is the interaction (not the edge Jaccard of Sec. 3) $J = [\Delta(\mathrm{SFT}, D_{\mathrm{SFT}}) - \Delta(\mathrm{SFT}, D_{\mathrm{base}})] - [\Delta(\mathrm{base}, D_{\mathrm{SFT}}) - \Delta(\mathrm{base}, D_{\mathrm{base}})]$, with a paired-prompt bootstrap CI and a within-prompt label-permutation p ($B = 10{,}000$ each). $J > 0$ would mean the SFT-specific sources matter more in SFT than in base, relative to the base-specific ones, so a main effect of "these features matter everywhere" cancels.

The pilot, and four defects found before the sweep. A stop-gated pilot ($k = 5{,}000$, 100 prompts) passed the gate (D_{SFT} moved the output 0.550/0.524 nats in base/SFT, 65×/90× its random control) and gave $J = -0.027$ (CI $[-0.055, -0.001]$, $p = 0.050$). An earlier draft read it as the causal result. Re-reading its inputs for the sweep found four defects. (i) The committed reader ended in a catch-all `else` that labelled every non-F1 outcome F2, although F2 requires a CI covering zero; the reader now has an exhaustive table (adding F5, reversed: $J < 0$, CI below zero, $p < 0.05$; and U, discordant: the CI and the permutation test disagree), and under it the pilot reads U. (ii) D_{SFT} came from the audited SFT run's graph, but the pilot's SFT arm loaded a *different* seeded SFT run's adapter. (iii) The random control was frequency-matched on one 64-token prompt, so features silent there were tied. (iv) The model was loaded with weight processing, unlike the graph builder. The correction (notes/PREREG_functional_transfer_full_20260924.md) was committed before any sweep receipt existed. It fixes the pairing (configuration A, the audited graph with its own adapter, is the only one read for the verdict), frequency matching (firing rate over the 500 graph-building prompts, draws within ±200 ranks, without replacement), the held-out set (every row 5,000–9,999 of `pile-10k` with ≥ 64 tokens, $n = 4{,}816$, count committed before the main job) and the rule across budgets (one branch at all three k, otherwise MIXED).

Table 12: **Registered sweep ($n = 4{,}816$ held-out prompts; sweep verdict: MIXED).** A is the audited SFT graph with its own adapter (the only configuration read for the verdict); †descriptive: B is a second matched SFT run (seed 0), C the pilot's mismatched pairing at full n. Δ: $\Delta(\mathrm{SFT}, D_{\mathrm{SFT}})$ in nats; ratios are mean Δ over the matched random control in the SFT cell (D_{SFT}, SFT model) and the base cell (D_{base}, base model); a base-cell ratio below 1 fails the gate (F3), and then J is not read. Receipts `notes/functional_transfer_full_{A,B,C}_k*.json`, read only by `functional_transfer_sweep_read.py`.

cfg	k	$\lvert D_{\mathrm{SFT}}\rvert/\lvert D_{\mathrm{base}}\rvert$	Δ	SFT cell	base cell	J	95% CI	p	branch
A	2,000	55/58	0.259	37×	0.3×	—	—	—	F3
A	5,000	104/97	0.549	22×	2.0×	−0.0271	[−0.0326, −0.0217]	0.0001	F5
A	10,000	213/206	0.373	12×	7.9×	−0.0164	[−0.0217, −0.0113]	0.0001	F5
B†	2,000	50/57	0.272	27×	0.5×	—	—	—	F3
B†	5,000	116/106	0.610	14×	0.5×	—	—	—	F3
B†	10,000	225/212	0.448	6.5×	5.5×	−0.0221	[−0.0278, −0.0165]	0.0001	F5
C†	5,000	104/97	0.548	22×	2.0×	−0.0279	[−0.0336, −0.0225]	0.0001	F5

Result. The SFT-specific sources are load-bearing at every budget and in both models, 12–37× their random control in SFT (Table 12). At 10,000 edges the ratio of medians in the SFT cell is 3.5× (the table reports ratios of means). Where both gates pass, the interaction is *negative* and resolved: relative to the base-specific sources, the SFT-specific ones matter slightly less in the SFT model than in the base. At 2,000 edges the 58 base-specific sources move the base model's output less than their random control, so that budget is F3. The registered rule therefore returns MIXED (F3, F5, F5), and no stage-specificity claim is made in either direction. The second matched SFT run (B) fails the base-cell gate at 2,000 and 5,000 edges and has the same negative sign at 10,000. The pilot's pairing at full n (C) lies within 0.001 nats of A in every cell ($J = -0.0279$ vs. -0.0271), so the pairing defect did not create the sign. The drop from the pilot's 65× lies in the control: at $k = 5{,}000$ the

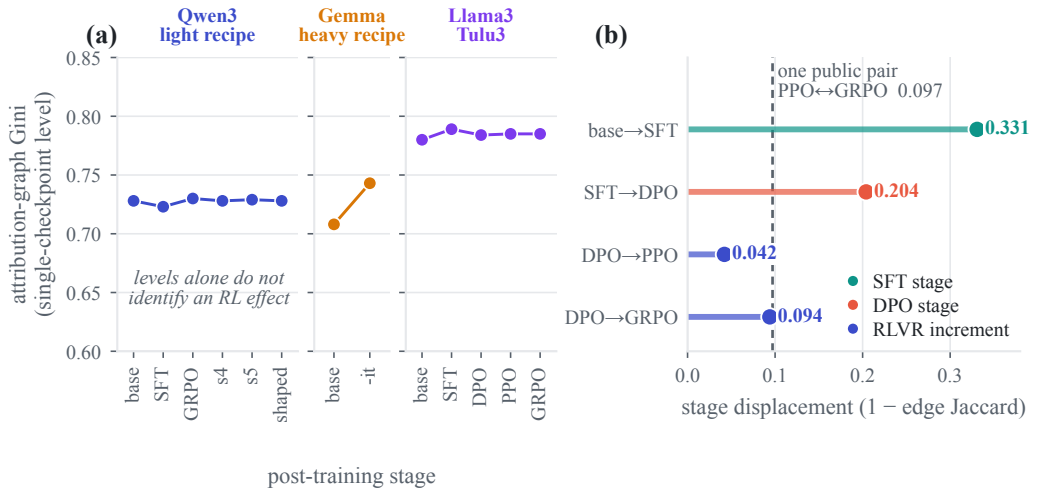


Figure 7: **Checkpoint levels and stage increments answer different questions.** (a) Co-activation-graph Gini for Qwen3-1.7B light recipes, Gemma-2-2B heavy instruction tuning, and the Llama-3.1-8B Tulu-3 sequence. Levels alone do not identify an RL effect. (b) Stage displacement is 1 minus edge Jaccard. The dashed line is one public PPO/GRPO pair (0.097), not a same-recipe seed-null distribution. The RLVR increments are unresolved against that single reference; they are not certified nulls. Receipts: `notes/basediff_{qwen,gemma}_summary.json`, `tulu_basediff_summary.json`, `tulu_stage_increments.json` (v1); generator `code/analysis/fig_staircase_money.py`.

base-model random control rises from 0.008 to 0.027 nats with the corrected frequency matching, loader and prompt set (their shares are not separated), while $\Delta(\text{base}, D_{\text{SFT}})$ moves from 0.550 to 0.581. *Wording deviation, declared.* For a MIXED verdict the correction's pre-written sentence was "found the displaced sources load-bearing ... but did not resolve stage specificity". Because two of the three primary budgets are resolved F5 ($p = 0.0001$), we state the direction instead (abstract, Introduction, Sec. 4.2, Discussion, Conclusion); the claim that follows is unchanged. The verb stays *displaces* under every registered verdict: this source-side intervention corroborates structure but does not validate the co-activation object as an executable circuit.

L TULU OPERATING-POINT ROBUSTNESS AND SAE-FAMILY CAVEATS

`Llama-Scope` SAEs are dataset-wise normalized: per-layer average activation norms are 12.1/31.6 at l16/l25, and the JumpReLU threshold lives in normalized space, so a direct `sae_lens` `encode()` call runs unnormalized unless the scale is folded into the weights. We therefore ran the full staircase twice, unnormalized (v1) and normalization-folded (v2). Every conclusion is operating-point robust. The post-SFT stage increments agree to within 0.001 across the two runs (SFT→DPO 0.796, DPO→PPO 0.958, DPO→GRPO 0.906, PPO↔GRPO 0.903); the base→SFT increment, the largest-displacement and most operating-point-sensitive stage, moves from 0.669 (v1) to 0.697 (v2) — still by far the dominant displacement stage, so the qualitative conclusion is unchanged; the DPO→PPO increment is the one that straddles a rounding boundary (v1 0.9579 / v2 0.9587), and we quote the v1 value 0.958 to match the main table; the run-pair null PPO↔GRPO is 0.9029 (v1) / 0.9033 (v2), reported as 0.903; at the v2 point level deltas stay $\leq$ 0.019 against a 0.815 base Gini (v1: $\leq$ 0.009 against 0.780, Table 2); the battery verdict is unchanged. The Llama-3.1-8B LoRA-SFT replication (App. E) was run at the normalization-folded (v2) point. Two caveats are grouped here, with full treatment in Limitations. First, `Llama-Scope` runs far denser than `qwen-scope` (measured $L_0 \approx 1.5$k of 131k at the normalized point, vs. 50), so absolute L_0 does not compare across SAE families; we only ever use within-family deltas. Second, we exclude reconstruction-error (FVU) numbers for `Llama-Scope`: its `sae_lens` *decode* path returns im-

plausible error ($\sim$10,000$\times$ variance, against the $\sim$90% FVE its authors report), indicating a loader inconsistency, and everything in this paper uses `encode` only.

M H2 / H3 AUXILIARY-NULL DETAIL (V0 PYTHIA-70M PANEL)

H3 in full: non-collapse, consistent with a barely-perturbed basis. H3 is a *binary* non-collapse test: does the per-feature entropy delta clear the $|d| = 0.5$ collapse gate (a $|d| = 0.5$ threshold we pre-registered ourselves, following Cohen's conventional medium effect; Yue et al. (2025); Wang et al. (2025) report behavioral results and state no feature-level gate, so this bar is ours to defend, not theirs)? Across the committed accuracy-only GRPO runs the answer is *no*: the cross-tokenizer run (24 probes $\times$ 4 continuations, `h3_entropy_cross_tokenizer_v0.json`) gives $\Delta H = -0.004$, $d = -0.005$; re-measuring the same baseline checkpoint at three probe$\times$continuation budgets gives $\Delta H = +0.018$, $d = +0.26$ (24$\times$4), $\Delta H = +0.043$, $d = +0.32$ (24$\times$1), and $\Delta H = -0.020$, $d = -0.19$ (96$\times$1) — small, sign-unstable, and all far below the gate, so we report only the binary non-collapse. A *reward-shaped* regime (shaped-λ0.1, which adds an SAE-feature shaping term) shifts the delta to $\Delta H = -0.024$, $d = -0.29$ (still sub-gate): one reward-shaped run moves H3 slightly more than the accuracy-only baseline, within the spread of our probe-budget reruns, so we draw no conclusion about reward structure from it. On the committed cross-tokenizer run the largest per-feature entropy deltas are heavy-tailed and *include* two of the four v0-panel top-attribution features (#8286 and #32255, both $\Delta = +0.11$, among the top-5 gainers, and a third, #9666, tied with them at $+0.110$ and left out of the receipt's top-5 list only by tie order; Table 14); the aggregate delta is nonetheless negligible ($d = -0.005$).

Pretraining-crystallized directions. The base-stable concentration admits a pretraining-crystallized reading: Moskvoretskii et al. (2026) find persona vectors (Chen et al., 2025) form within 0.22% of OLMo-3 pretraining and, extracted that early, still steer the post-trained instruct model, although they keep refining through pretraining. Our scale-stable top-1% concentration and the per-destination-layer single-hub slots of the 8B scan (#35239@22, #15675@26, #21438@30, #12735@32) are compatible with that reading, but the 8B graphs have no SFT or GRPO checkpoint, and identical integer indices across layers are not aligned features. Per Gerasimov et al. (2026) the reproducible unit is a *subspace* not a literal index; the two withdrawn v0 coincidence claims are detailed below.

Panel provenance and measurement space. H2 and H3 audit the real `Qwen3-1.7B` SFT/GRPO checkpoints, but *indirectly*: each checkpoint generates text, and a frozen external reader — Pythia-70m with the `pythia-70m-deduped-res-sm` residual SAE, the v0 pilot pair — encodes the generated text (protocols in App. D). The 32-feature panel was preselected by a function-word-outcome screen — features ranked by association with the next token falling in a closed set of function words, over 2,048 rollouts at sequence length 64 (positive rate 0.092); the selection receipt, including all 32 indices and the screen's definition, is committed at `notes/panel_v0_pythia70m.json` and subsequently evaluated on the v0 pilot graph in this same reader space, so the measurement is well-defined: panel indices (e.g. #32255, #9666, #8286, #5158) index the *reader's* SAE dictionary, which is identical across checkpoints because the reader never changes; only the generated text differs between pre and post. These indices are therefore unrelated to the `qwen-scope` H1 hub indices (#30329, ..., Table 6), and H2/H3 probe the checkpoints' *generated prompt-plus-continuation text* through a fixed SAE lens rather than their internal activations — auxiliary, generation-side corroboration, not an internal-activation null.

H2 (activation delta). The committed cross-tokenizer receipt (`h2_diff_v0_cross_tokenizer.json`) is the shaped-λ0.1 arm ($n = 16$ probes; pre = SFT, post = shaped-reward GRPO). On the v0 panel's top-32 features its per-feature mean-activation delta lies in $[-0.008, +0.006]$ (weighted cosine 0.9998, Jaccard 1.0, identical 16/16 active counts). The four top inbound hubs of the v0 pilot graph (#32255, #9666, #8286, #5158 by inbound weight in `h1_summary_v0.json`; Table 13) all move in the *negative* direction ($-0.0064, -0.0060, -0.0040, -0.0003$ respectively), consistent with this arm's reward explicitly penalizing the panel's activations; no feature is switched off. The cosine-near-1 is not a precision floor: delta SD 0.0029 is $\sim$ 5% of the 0.0518 mean magnitude (Table 13, Fig. 9). (An earlier

accuracy-only baseline-GRPO H2 run is not among the committed receipts, so we report only the shaped arm here.)

Table 13: **H2 top-8 per-feature mean-activation deltas (committed shaped-$\lambda 0.1$ arm, cross-tokenizer, `h2_diff_v0_cross_tokenizer.json`, $n = 16$ probes).** Under this reward-shaped arm three of the four v0-graph inbound hubs (`h1_summary_v0.json`) are among the top-8 movers, all *negative* (the fourth, #5158, moves -0.0003); the same 16 of the 32 panel features stay active (weighted cosine 0.9998, Jaccard 1.0).

Feature ID	Mean act. (pre)	Mean act. (post)	Δ	In hub set?
14907	0.0248	0.0168	-0.0079	no
32255	0.4790	0.4727	-0.0064	yes
8232	0.0047	0.0110	$+0.0063$	no
9666	0.4526	0.4466	-0.0060	yes
32257	0.0103	0.0155	$+0.0053$	no
12234	0.0096	0.0142	$+0.0046$	no
27648	0.0869	0.0826	-0.0043	no
8286	0.3018	0.2978	-0.0040	yes

H3 (entropy). The probe-sensitivity calibration: injecting a known entropy change via sampling temperature ($0.5 \to 1.5$ gives $d = +0.18$; the extreme $0.3 \to 2.0$ gives $d = +0.21$) shows the metric is directionally correct; we therefore read H3 only as a binary non-collapse test rather than a calibrated effect-size instrument. A stronger positive control via a Wang et al. (2026b) injected-bias checkpoint is pre-registered. The largest per-feature entropy deltas are heavy-tailed (Table 14); two of the four v0-panel top-attribution features (#8286, #32255) are themselves among the top-5 gainers ($\Delta = +0.11$), and a third (#9666) ties the fifth at $+0.110$, excluded from the top-5 list only by tie order, so the entropy movement is not confined to low-attribution features.

Table 14: **H3 per-feature entropy deltas (top-5 gainers + top-5 collapsers), committed cross-tokenizer run (`h3_entropy_cross_tokenizer_v0.json`, v0 Pythia-70m panel).** Two of the four v0-graph inbound hubs (#8286, #32255) are among the top-5 gainers ($\Delta = +0.11$); a third (#9666) ties the fifth row at $+0.110$ (four features share this value; the receipt's tie order drops it); the aggregate delta is nonetheless negligible ($d = -0.005$, gate not passed).

Feature ID	Entropy (pre)	Entropy (post)	Δ	In hub set?
14907	0.980	1.151	$+0.171$	no
22117	1.812	1.941	$+0.130$	no
8286	1.812	1.922	$+0.110$	yes
32255	1.812	1.922	$+0.110$	yes
2180	1.812	1.922	$+0.110$	no
9522	0.794	0.616	-0.178	no
28334	0.354	0.260	-0.095	no
12234	0.995	0.659	-0.335	no
8232	0.747	0.345	-0.401	no
3201	1.718	1.205	-0.513	no

Withdrawn v0 coincidence claims. On the v0 Pythia-70m panel the four highest-attribution features also had the highest edge in-degree (a v0-pilot small-cardinality observation that does *not* transfer to the qwen-scope H1 graph). We withdraw two earlier-draft claims: (1) the "entropy-widening" fourth dimension (the $+0.206$ hub-widening first measured on a smaller probe set is not reproduced on the committed cross-tokenizer run, where three of the four v0 top features move by $\approx +0.11$ and the fourth (#5158) by $+0.06$; Table 14); (2) the "$4/4$ pass the Steering $\sigma_{\text{KL}} \geq 2$ gate" arm and its $1/35{,}960 \approx 0.000028$ significance — a corrected steering analysis, computed on σ_{KL} (not the clamp denominator σ_f), clears $0/19$. The same-backbone 8B coincidence (#35239; L4, App. F) does not depend on any steering result and is retained.

Base-control visualization. Figure 3 plots the base→post Gini change for every post-training checkpoint (Table 5): all five light-recipe Qwen runs sit inside the $|\Delta\text{Gini}| \leq 0.005$ observed en-

velope, only heavy `gemma-2-2b-it` clears it (+0.035), and even that is an order of magnitude below the single-checkpoint Gini level.

Layer-local hub-slot diagnostic. Figure 8 scans a 15-hub panel across 31 destination layers of the `Qwen3-8B` widekt graphs: 13 of the 15 appear in at least one graph's top-30 inbound set; only three ids recur at more than one destination layer (#28040 at l12–l19, #21438 at l30–l31, #58220 at l21/l34), and because each destination layer has its own independently trained SAE, a cross-layer index match is not feature identity; the terminal-l35 cell is unembedding-dominated; these slot statistics do not establish cross-layer feature identity.

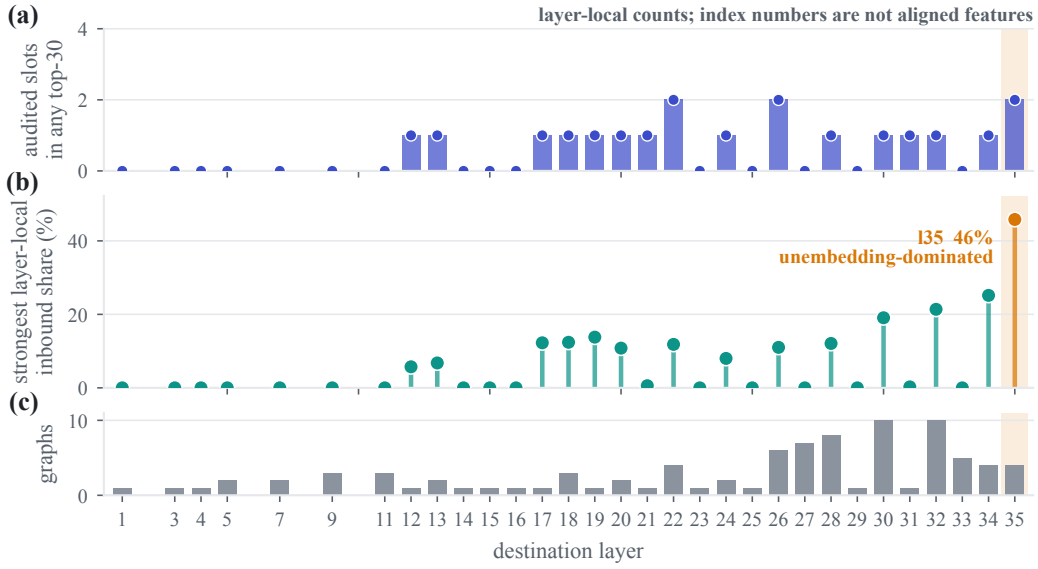


Figure 8: **A layer-local diagnostic of the audited index slots.** (a) Count of the fifteen audited integer slots present in any top-30 set at each destination. (b) Their maximum inbound share within that destination. (c) Number of available source graphs. The SAE dictionaries were trained independently by layer: identical integer indices are not aligned features. This plot therefore reports local slot statistics, not cross-layer feature recurrence or preserved feature identity. Layer 35 is unembedding-dominated. Receipt: `notes/fig8_backbone_recurrence_data.json`; generator `code/analysis/fig8_layer_index_diagnostic.py`.

H2 delta diagnostic figure. Figure 9 shows the per-feature pre- vs. post-GRPO mean activations (panel a) and the histogram of post−pre deltas (panel b) for the top-32 features of the v0 Pythia-70m panel, on the *shaped-reward* (shaped-$\lambda 0.1$) arm. The four top-attribution features (highlighted) sit on the $y = x$ diagonal with small *negative* deltas — the reward explicitly penalizes this panel's activations — and are polarity-preserving redistributions, not on-off switches.

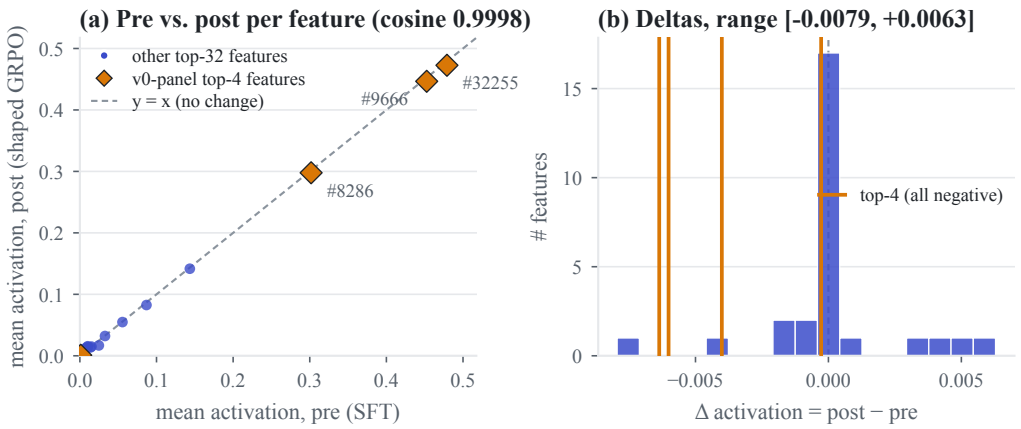


Figure 9: **H2 per-feature delta diagnostic (cross-tokenizer, $n = 16$ probes).** (a) Per-feature mean activation pre (SFT) vs. post (shaped-reward GRPO, shaped-$\lambda 0.1$ arm), 32 features. v0-panel top features (amber diamonds) sit on the diagonal with small *negative* deltas, consistent with this arm's reward penalizing the panel's activations. (b) Histogram of post−pre deltas; the four v0-panel top features are marked. Cosine 0.9998 is consistent with the delta magnitudes shown ($\sim 5\%$ of pre-activation mean), not numerical precision.