BENCHMARKING INTERPRETABILITY LOCALIZERS FOR ADAPTATION PLACEMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Parameter-efficient fine-tuning makes it cheap to adapt a language model to a shifted distribution, but not to decide *which layers* should receive the update. Mechanistic-interpretability localizers—sparse autoencoders, attribution patching, gradient and Fisher saliency, drift-native gradient directions, linear probes—promise "where," and that promise is a falsifiable prediction: if a localizer identifies where the shifted computation lives, adapting those layers should beat adapting anywhere else at the same budget. We build the ground truth to test it: dense grids of matched-budget LoRA adapters at every candidate layer window, across four domains and three model families (Gemma-2 9B; Qwen-2.5 7B and Llama-3.1 8B), each grid measuring which placements actually recover the task. Scored against the grids under a frozen protocol, seven localizers—including AdaLoRA, a method built to allocate adaptation budget—yield one verdict: **no localizer reliably predicts where to adapt.** Where placement is certifiably flat, every localizer ties a random matched-budget window; where placement demonstrably matters, even the localizers that name the peak confer no certified advantage over a random window; a depth band frozen on two model families breaks on the third; and no windowed placement, localizer-chosen or optimal, recovers whole-network performance. The negative is bounded, not merely unresolved: no tie conceals a localizer advantage above $+5.7$pp (95% confidence). We release **PlacementBench**—the grids, the frozen scoring-and-admissibility protocol, and effect bounds on every tie—as an instrument rather than an indictment: a future localizer overturns this result by clearing its two bars, identification and certified advantage. A first use: a cross-cell readout calibrated on the graded grids identifies a top-2 window in $8/9$ cells and clears the certification bar on one, where every per-cell rule fails.

1 INTRODUCTION

Mechanistic interpretability makes a falsifiable prediction about model adaptation. If a localizer identifies *where* a model represents or processes a shifted computation, then adapting those layers should recover the task better than adapting anywhere else at the same budget. It is rarely stated as a prediction, but it is the premise of turning any interpretability signal into a layer recommendation—and it is directly testable.

Parameter-efficient fine-tuning (PEFT) makes it cheap to adapt a large language model to a shifted deployment distribution (Hu et al., 2022), but it does not answer the targeting question: when only a fraction of the network will be updated, *which layers should it be?* If interpretability can point to the right layers, targeted placement promises whole-network recovery at a fraction of the cost. A practitioner today has a substantial menu of localizers to ask: sparse autoencoders and direct-logit attribution (Lieberum et al., 2024; Bricken et al., 2023), attribution patching and its corrections (Kramár et al., 2024), gradient-based saliency (Lee et al., 2023) and Fisher information (Kirkpatrick et al., 2017), gradient-direction signals (Kamat et al., 2026), and linear probes (Alain & Bengio, 2016). The premise is not hypothetical: Wang et al. (2025) allocate LoRA rank by identifying the layers with the largest circuit-edge changes, Lee et al. (2023) select layers to fine-tune under distribution shift via gradient-based criteria, and practitioners routinely adapt circuits surfaced by SAE attribution (Lieberum et al., 2024) or attribution patching (Kramár et al., 2024). **None has been evaluated against ground-truth placement at window granularity on LLMs.**

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.

Every parameter-budgeted adaptation makes a placement decision, implicitly or explicitly: layer-selective fine-tuning is standard practice for cutting memory and compute (Pan et al., 2024; Lee et al., 2023), and adaptive methods allocate budget across layers as their core mechanism (Zhang et al., 2023). Interpretability's action-guiding promise is that knowing *where* a capability lives should inform intervening *there*, and budgeted placement is the cheapest intervention to test. If localizers cannot beat costless baselines here, that is news about the promise itself, not about a niche task.

This paper supplies that ground truth and runs the evaluation. For each domain we construct a *placement grid*: a dense set of matched-parameter LoRA adapters, each unfreezing a different contiguous layer window, all trained identically on the shifted task. The grid directly measures what every localizer tries to predict, turning "where to adapt" into a benchmarked quantity. We then score seven localizers by mapping each one's output to a predicted window under a single frozen rule and comparing to the grid. Figure 1 (§4.2) shows the measured placement landscapes with every localizer's predicted window overlaid.

Findings. The result is a systematic negative, and the negative is the contribution. It is also a *bounded* negative: every contrast family was frozen before unblinding, no tabulated tie conceals a localizer advantage larger than $+5.7$pp (one-sided 95% upper bound), and the tests had 80% power at 4.3–7.5pp—the size of effects the grid *did* certify (Appendix A.14).

1. **No localizer reliably predicts placement.** Where the grid is *flat* (code; regex at 84.7pp headroom), every localizer, a random matched-budget window, and the plateau optimum are indistinguishable (exact McNemar), a tie surviving re-seeding, ten draws, doubled capacity (§4.5). Where it is *peaked* (JSON), $0/7$ clear both bars: three are hook-capped (their reading hook sits below the peak; vocabulary: Appendix A.1), the two that name it gain no certified advantage over the best random window, and sweeping the hook across all 42 layers rescues none (§4.3). Two fixed-depth prescriptions—a zero-cost heuristic that tops the Gemma-2 battery, and a band frozen from two architectures—both fail out-of-sample on Llama (§5).
2. **Reliability $\neq$ validity.** Split-half stability and predictive accuracy dissociate in both directions: the battery's most reliable localizer names the identical (wrong) window in every domain and every half—*reproducibly wrong*—while SAE-131k's window flips one grid step under resampling even as its top-5 layers agree $5/5$.
3. **The tooling ecosystem is patchwork.** No single localizer toolchain covers any model in the study: SAE and tuned-lens coverage are nearly disjoint across families, and of Llama Scope's two SAE widths only $32\times$ passes the construct-validity gate (§5).
4. **A headroom-admissibility protocol.** Not every domain can grade localizers: every attempted domain reports headroom H with noise statistics z and R (§3.4). Factual QA's headroom is indistinguishable from zero on all three architectures ($H = 2.7$, 1.1, and 3.8pp; $z \leq 0.8$), and we show why (§4.7).
5. **Targeting never recovers the ceiling.** In every resolved domain, no windowed placement—localizer-chosen or optimal—recovers whole-network performance, and every paired gap is significant (§4). A matched-cost control makes it a prescription: on Llama code (both generations), spreading the best window's budget across *all* layers beats it outright (Llama-3 $p = 0.0003$; Llama-3.1 $p = 2.4\times10^{-9}$) and ties the ceiling at $\sim$22% of its parameters (Appendix A.12).

What this negative does—and does not—claim. Of the seven, only **AdaLoRA** was designed for placement—allocating budget across layers is its core mechanism; the other six we do not score against their design goals. The claim is about a *decision*: when a practitioner must place a fixed budget, none beats baselines that cost nothing (a random matched-budget window, a fixed depth fraction, or the whole network). The grid grades two axes: *identification* (naming a top-2 window) and *certified advantage* (beating matched random draws under correction). Both bars are winnable. AdaLoRA cleared identification on Gemma-JSON by naming the optimum; one Qwen-regex window cleared certified advantage (caveats: Appendix A.10). But no localizer clears both on any single landscape—that conjunction is the benchmark's bar, and clearing it per cell overturns this result (§5 tests an eighth, cross-cell rule).

2 Related Work

Sparse autoencoders and feature attribution. SAEs decompose activations into sparse, often interpretable features (Bricken et al., 2023; Cunningham et al., 2023), motivated by superposition (Elhage et al., 2022). Gemma Scope (Lieberum et al., 2024) provides pre-trained SAEs across Gemma-2 layers and dictionary widths, which we use as two of our localizers (16k and 131k). Activation/attribution patching (Conmy et al., 2023; Kramár et al., 2024) attributes behavior to components; we adapt the GradDrop correction of AtP* (Kramár et al., 2024) to per-layer residual nodes. Prior SAE work targets model diffing and interpretation (Lindsey et al., 2024), not adaptation placement.

Parameter-efficient and layer-selective adaptation. LoRA (Hu et al., 2022) trains low-rank updates; AdaLoRA (Zhang et al., 2023) and SoRA (Ding et al., 2023) allocate rank, and gradient-magnitude heuristics select layers (Lee et al., 2023). These prescribe *how much* to adapt per layer but are candidate localizers for *where*; our Fisher-information and gradient-direction localizers represent this family. Lee et al. (2023) is the closest prior work—block selection validated by per-block brute force—positive but coarse-grained, mostly vision shifts; at window granularity on decoder-only LLMs this family never separates from random (§4).

Mechanistic interpretability for adaptation. Wang et al. (2025) use circuit analysis to allocate LoRA rank. Our question is prior to theirs: not how to distribute a budget once a region is chosen, but whether any localizer identifies the right region at all, measured against a ground-truth grid. The SAE localizers are the softest fit, included because practitioners adapt SAE-surfaced circuits regardless. Meng et al. (2022) and Geva et al. (2023) establish that factual knowledge is distributed across MLP layers, bearing on our QA domain. Hase et al. (2023) find causal-tracing localization does not predict which layers edit best; our grids find the same read/write dissociation for gradient-based adaptation.

Probing and lenses. Linear probes measure whether a property is linearly decodable at a layer (Alain & Bengio, 2016; Belinkov, 2022); we use an emergence-layer probe as a localizer. The tuned lens (Belrose et al., 2023) decodes intermediate representations toward the vocabulary; its coverage (Qwen-2.5, not Gemma-2) feeds the portability finding.

Concurrent gradient-signal work. Kamat et al. (2026) (Grad Detect) use gradient *direction*, not magnitude, to predict hallucination risk. We adopt a gradient-direction localizer in their spirit (comparison: Appendix A.7).

3 PlacementBench

We benchmark localizers against a ground-truth answer to one question: given a model $\mathcal{M}$ fine-tuned on distribution T_1 and facing shifted distribution T_2, which contiguous layer window, at a fixed LoRA budget, best recovers T_2 performance? We build the ground truth (§3.1), define the localizer battery (§3.2), fix the scoring rule (§3.3) and two grade-admissibility criteria (§3.4).

3.1 Ground-truth placement grid

For each domain we train a dense grid of LoRA adapters that differ *only* in which layers they unfreeze, holding data, steps, rank, and trainable parameter count fixed within a budget class. The grid (23 arms per domain at 9B) tiles the layer axis with: matched-budget windows at centers $\{14, 18, 22, 24, 26, 30, 34\}$; an offset ladder across depth; whole-network and early-layer controls; and null baselines: *random* windows and a gradient-selected window. Each arm's held-out T_2 accuracy is the ground truth for that placement.

3.2 Localizer battery

The primary battery runs on Gemma-2 (Gemma Team, 2024) (9B, 42 layers, 4-bit NF4); §5 extends the portable subset to Qwen-2.5 7B and Llama-3.1 8B, where a pretrained SAE (Llama Scope) re-enters the battery. The seven localizers are the practitioner's menu at layer granularity:

1. **SAE-16k** — direct logit attribution of the T_2–T_1 feature-activation differential onto *birthed* SAE features (high on T_2, near-zero on T_1; Gemma Scope, 16,384 features, layer 20).
2. **SAE-131k** — identical attribution at $8\times$ dictionary width, a width-robustness check.
3. **AtP+GradDrop** — attribution patching with the GradDrop correction (Kramár et al., 2024), adapted to per-layer residual nodes with a mean-ablation reference (the AtP* query/key fix is inapplicable at layer granularity; Appendix A.2.1).
4. **Fisher** — per-layer sum of squared gradients of the T_2 loss w.r.t. the base attention projection weights (the honest gradient-magnitude localizer).
5. **Gradient-direction** — per-layer $1 - \cos(\nabla L_{T_2}, \nabla L_{T_1})$, a drift-native direction signal after Kamat et al. (2026).
6. **Linear probe** — per-layer logistic probe discriminating T_1 vs. T_2 activations, scored by *emergence layer* (first layer at 90% of cross-layer maximum).
7. **AdaLoRA** — adaptive rank allocation (Zhang et al., 2023) as a *trained* localizer: a fresh all-layer adapter (q/k/v/o, average rank 1) trained from the same base$+T_1$ function on the same T_2 data, epochs, and sequence length; its signal is the retained per-layer rank after pruning (protocol and certification in Appendix A.4).

The first three localizers read the residual stream at the SAE hook layer (20) and are *hook-anchored*: their window cannot exceed the hook; Fisher, gradient-direction, the probe, and AdaLoRA span the whole network; the asymmetry is itself a result (§4).

3.3 Scoring: from localizer to predicted window

Every localizer emits either a per-layer profile or a single emergence layer. A single frozen rule maps this to one grid window, fixed *before* any profile is computed (no per-method tuning). For a profile P, let $B(P)$ be its maximal contiguous run at $\geq 50\%$ of $\max P$ (layer 0 excluded); then

$$\widehat{W}(P) = \mathrm{snap}\big(\max B(P) + 2\big),$$

where snap maps to the nearest grid center (ties resolved downstream) and the probe's emergence layer substitutes for $\max B(P)$. A diffuseness guard grades profiles spanning over half the network "failed to localize"; a coverage guard flags off-grid predictions.

Worked example (Fisher, Gemma-2 JSON). Fisher's profile points at layer 25; the frozen rule maps this to the window centered at 26. That window scores 0.812, statistically tied with the grid's best (c24 = c26 = c30; optimum c30 at 0.840)—an *identification* hit. The second bar is graded against the costless alternative: the best random matched-budget draw scores 0.796, and not even the grid optimum's edge over it survives correction ($p = 0.019$, raw-significant only; §4.4). Naming the peak region bought nothing a random draw did not—the pattern the battery repeats.

3.4 Admissibility of a grade

Two criteria determine whether a grade is meaningful.

Headroom statistics. Let $H = p_c - p_f$, where p_c is whole-network (uniform-all) held-out accuracy, p_f is the no-T_2 floor (base + T_1 evaluated on the same T_2 held-out set), and n is the shared eval size. We report two noise-relative statistics alongside H:

$$z = \frac{H}{\sqrt{p_f(1-p_f)/n + p_c(1-p_c)/n}}, \qquad R = \frac{H}{\sqrt{p_c(1-p_c)/n}}.$$

z (a conservative unpaired difference-of-proportions statistic) asks whether any headroom exists at all; R expresses the headroom in units of a single sweep point's error bar—an upper bound on how many distinguishable accuracy levels the placement landscape could contain. Both are reported continuously, with no threshold: a domain whose H is within noise ($z \lesssim 1$) grades *uninformative*—not a null for any localizer, but insufficient signal to grade any—and individual placement claims are gated per-contrast by paired McNemar tests (§4.4), never by H itself (vocabulary: Appendix A.1).

Table 1: Predicted adaptation window per localizer vs. ground-truth grid optimum, Gemma-2 9B, code and JSON domains. A localizer *hits* if its window is in the tie-inclusive top-2 of the accuracy curve after snapping to the nearest center (§3.3; snapping detail in Appendix A.2.3); DIFFUSE = band spans >half the network, no window named. Hook-anchored localizers (SAE-16k/131k, AtP+GradDrop) are capped at grid center 22 by the layer-20 SAE hook (§4.3). Depth heuristic and a random window for reference. QA ungraded ($H = 2.7$pp; §3.4).

Localizer	Code (opt. c24, flat)	JSON (opt. c30, peak)	Family
SAE-16k	DIFFUSE	c16→c18 ×	hook (≤c22)
SAE-131k	c18 ×	c18 ×	hook (≤c22)
AtP+GradDrop	c14 ×	c14 ×	hook (≤c22)
Fisher	DIFFUSE	**c26** ✓	whole-net
Gradient-dir.	c38 (off-grid) ×	DIFFUSE	whole-net
Linear probe	c14 ×	c14 ×	whole-net
AdaLoRA	c34 ×	**c30** ✓ **(= opt.)**	whole-net (trained)
Depth 67/75%	*c30* ✓	*c30* ✓ *(= opt.)*	zero-cost
Random window	*tie (best* 0.520*)*	*tie (best* 0.796*)*	baseline
Grid optimum	*c24 (*0.510*)*	*c30 (*0.840*)*	truth

Three-cause flatness rule. A flat grid can arise from (i) genuine placement-invariance, (ii) insufficient headroom resolution (low R), or (iii) broken training; it grades as invariance *only* if headroom is present and training is healthy (loss descended over real steps, non-zero NLL change). This prevents reading a degenerate run as a scientific null.

4 RESULTS

4.1 THE PLACEMENT LANDSCAPES DIFFER SHARPLY BY DOMAIN

Code: flat. The code grid is flat: matched-budget windows span only 0.423–0.510 held-out accuracy, and random windows, a gradient-selected window, and the plateau optimum are indistinguishable (§4.4). Headroom is ample ($H = 33.3$pp, $z = 8.8$, $R = 11.7$: whole-network reaches 0.580 over the 0.247 no-T_2 floor, $n = 300$), so the flatness is informative. Placement does not matter at this budget; the only structure is that early/detection layers are worse ($p < 0.001$, Bonferroni-surviving). A gradient-selected window ties the plateau optimum and all three draws ($p = 0.71$; 1.00/0.80/0.06), falsifying gradient saliency here.

JSON Schema: peaked. The JSON grid is sharply peaked: across the matched-budget family, held-out accuracy runs from 0.720 at the detection window (c14) to a 0.840 optimum at c30, and random windows alone span 0.688–0.796 (10.8 points). Headroom is the largest in the study ($H = 64.8$pp, $z = 19.6$, $R = 36.2$: whole-network reaches 0.912 over the 0.264 no-T_2 floor, $n = 250$). This is the domain where placement *demonstrably matters* (§4.2).

QA: uninformative—headroom within noise. The factual-QA grid is flat, headroom indistinguishable from zero: whole-network adaptation reaches $H = 2.7$pp over the no-T_2 floor ($z = 0.6$, $R = 0.8$ at $n = 183$), so ideal placement cannot move accuracy beyond one error bar; per the three-cause rule, QA grades *uninformative*, not placement-invariant (§4.7).

4.2 NO LOCALIZER RELIABLY PREDICTS PLACEMENT

Table 1 is the headline: each localizer's predicted window against grid ground truth, across code and JSON (regex flat, in prose below). Hit, miss, and DIFFUSE are defined in the caption; the optimum is code c24 (0.510, flat plateau) and JSON c30 (0.840, genuine peak).

Reading the table (Figure 1): on **code**, 0/7 localizers hit, a random matched-budget window ties the best of them (mean random-window regret 0.003, lower than every localizer's; the best draw (0.520) numerically tops the plateau optimum), and the zero-cost depth heuristic lands the plateau. On **JSON**, five of seven miss or abstain where placement matters: both SAE widths and AtP+GradDrop are

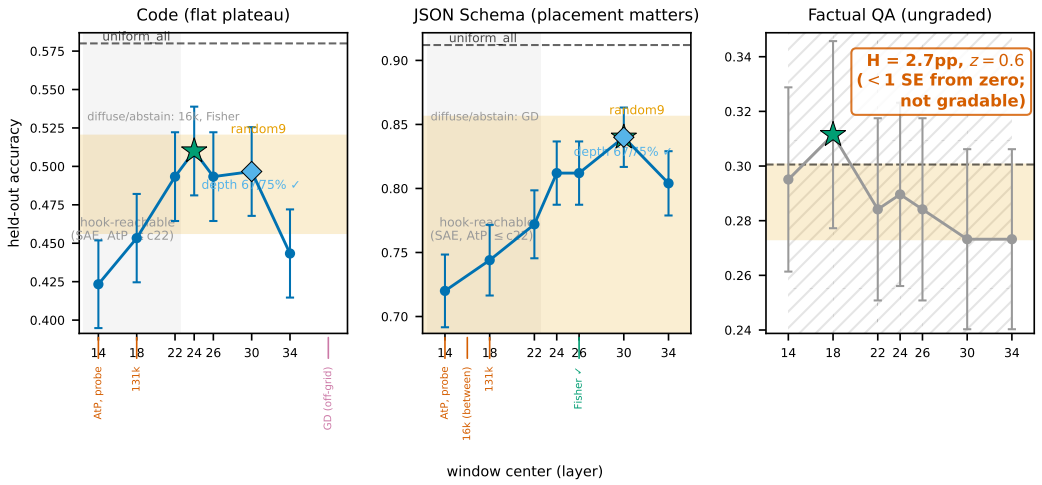

Figure 1: **Placement landscape (Gemma-2 9B): matched-budget (3.83 M parameter, 9-layer) held-out accuracy vs. window center, per domain.** Green star (⋆) = grid optimum; dashed line = UNIFORM_ALL full-budget ceiling; orange band = random-window min–max over three draws; error bars = ±1 binomial SE. Arrows = localizer-predicted centers (Table 1); ✓ = tie-inclusive top-two hit (§3.3). Gray shading = hook-reachable zone (≤c22); depth-heuristic baseline = labeled diamond; AdaLoRA not marked. **QA panel grayed, ungraded** (§3.4). The fourth domain (regex, flat) is reported in prose (§4.2).

hook-capped at c22 and cannot reach the c30 peak—a family ceiling, not a tuning miss (§4.3)—and the linear probe emerges at c14. Only Fisher and AdaLoRA land the peak region: Fisher's raw c25 scores as the c26 window, inside the c24=c26=c30 top-two tie; AdaLoRA's retained ranks name c30, the optimum—the only standard-setup localizer to do so. Yet the best draw (0.796) sits 4.4 points below the optimum and 1.6 below Fisher's window (0.812), and the frozen per-item exact-McNemar family (12 contrasts, Bonferroni $\alpha = 0.0042$) resolves the gaps. Placement itself is confirmed: the realized optimum separates from the detection window, early window, two of three random draws, and the gradient-selected window under correction (all $p \leq 0.0009$; vs. the worst draw the split is 38–0). But naming the peak confers no certified advantage: the frozen SAE window ties the best random draw ($p = 0.29$), as does the gradient-selected window ($p = 0.42$), and even the optimum's edge over the best draw is raw-significant only ($p = 0.019$, fails Bonferroni—and the optimum, an empirical max, is selection-inflated). AdaLoRA's window *is* c30, so its exact hit inherits no significance the window lacks (§4.4); Fisher-vs-best-draw was not frozen, left ungraded. **Even where placement demonstrably matters, no localizer's prediction demonstrably beats a random draw.**

We score under the tie-inclusive top-2 criterion, the reading that *helps* the localizers; under the stricter exact-top-2 criterion the battery goes $0/7$ on code and AdaLoRA's c30—the optimum, a hit under any criterion—is the sole JSON survivor. Nor does the verdict hinge on the band→window rule: *certified advantage* is tested on the window arms themselves, regret and the paired ties—the primary reading on flat landscapes—bypass the rule entirely, and every raw profile peak is reported alongside its snapped verdict. Criterion details and sensitivity checks: Appendix A.2.3.

A fourth domain: regex dialect. Regex (Table 10: $H = 84.7$pp, $z = 39$, $R = 53$, the largest Gemma-2 headroom) resolves *flat* and replicates the code pattern: the windowed maximum (c24, 0.860) is indistinguishable from all three random draws, the whole-network ceiling (0.917) beats the best window ($p = 0.006$), and the battery scores $0/7$ with the random baseline's mean regret ($+0.012$) beating every scored localizer. The one resolved structure is an early-worse region (c24 vs. c14, $p = 0.0059$)—and AtP+GradDrop, the battery's most reliable localizer elsewhere, names exactly that window. The untuned depth heuristic hits again (67–75% →c30, regret $+0.007$), its third top-2 in three resolved Gemma-2 domains. Per-localizer verdicts and exact draw p-values: Appendix A.17. (On Qwen-2.5 the domain resolves a raised late plateau instead; Table 2, §5.)

4.3 AN ALL-LAYER HOOK SWEEP: IS THE CAP THE WHOLE STORY?

The structural cap is a property of hook placement, not the attribution computation, so we measure that dependence directly: all three hook-anchored localizers are re-extracted at every residual-stream hook (0–41) and regraded under the frozen rule (Appendix A.3). For the SAE widths the verdict tracks the hook, not the signal: shallow hooks ($\leq$19) extrapolate into early windows (regret $+0.096$ to $+0.120$), hooks 27–31 land the c24/c26 tie ($+0.028$), and SAE-131k names c30 only at hooks 34/36; SAE-16k names c30 at *no* hook. But naming c30 inherits only c30's own edge over the best random draw—raw-significant only ($p = 0.019$, failing Bonferroni; §4.4)—so even the SAE family's best case never demonstrably beats random. AtP+GradDrop's verdict never moves: its mapped center (c3–c13) falls below the tested grid and clamps to c14 (extrapolated)—a $+0.120$ miss at all 41 gradable hooks. These cells are diagnostic only: Table 1 remains the verdict at the frozen layer-20 hook.

4.4 THE NEGATIVE RESULT SURVIVES PAIRED SIGNIFICANCE TESTING

Accuracy gaps could be an artifact of *which* items each adapter solves, so load-bearing comparisons use a paired, exact McNemar test (McNemar, 1947) on per-item correctness (per-item vectors certified against banked sweep accuracies). The 14 code and 12 JSON contrasts were fixed before any per-item result was seen, with per-domain Bonferroni correction ($\alpha = 0.05$; code $n = 300$, JSON $n = 250$); Table 5 (Appendix A.13) gives the load-bearing subset; Figure 2 plots every contrast. Appendix A.14 bounds every tie: no gradient-localizer advantage exceeds $+5.7$pp (code, any draw) or $+1.7$pp (JSON, best draw) at 95% confidence.

Two claims follow. First, the *bad* windows are genuinely bad: detection- and early-layer windows lose to the plateau optimum under Bonferroni—placement is not noise. Second, and decisively, **the gradient localizer is statistically indistinguishable from both a random matched-budget window and the plateau optimum**—the top of the code landscape is a *three-way statistical tie*. We do not claim the principled window "wins" (a random draw scores numerically higher); we claim no localizer separates from random.

4.5 THE TIES SURVIVE RE-SEEDING, MORE DRAWS, DOUBLED CAPACITY, AND BF16

Four controls each remove one alternative explanation for the ties: training seed, a lucky trio of draws, an adapter-capacity floor, 4-bit quantization. Re-training the twenty-eight load-bearing arms (Gemma-2, Qwen-2.5, Llama-3, Llama-3.1) under two additional *training seeds* gives seed standard deviations of 0.002–0.021, below the per-item binomial standard error; the placement and bridge separations, the Llama-3.1 anti-localization separation, and the code regret nulls replicate across seeds; JSON's optimum-ahead regret edge is direction-consistent in $3/3$ seeds (significant in $1/3$); the only non-replicators are the post-hoc winners' edges—code's best draw ($0.520 \rightarrow 0.493$ under both re-seeds) and the regex maximum's edge over its strongest draw ($1/3$)—the winner's-curse signature. Extending three draws to ten leaves the code optimum *inside* the spread while JSON's c30 clears all ten; doubled capacity (r=32, both architectures) and full precision (bf16, both graded Gemma domains) preserve every tie; the changed contrasts go the other way (detection-window separations that survive Bonferroni under NF4 do not at bf16). Non-significance is never read as equivalence; we claim only that none emerges. Protocols and full tables (Table 7): Appendix A.15.

4.6 RELIABILITY DOES NOT IMPLY VALIDITY

We add a reliability axis: each localizer's predicted window recomputed on two disjoint data halves (split-half; single frozen comparator, guard-enforced disjointness; AdaLoRA excluded, each replicate being a full training run). *Stability without validity*: AtP+GradDrop is the battery's most reliable localizer—the identical window (c14, the grid floor) in every domain and every half—and that window is wrong everywhere placement matters; on regex, c14 is the one window significantly below the plateau ($p = 0.0059$): *reproducibly wrong*. *Instability without a profile change*: SAE-131k's window flips one grid step under resampling *while its top-5 layers agree* $5/5$—whether a localizer looks stable partly reflects the band-reduction rule, not the localizer. On Qwen-2.5 and Llama-3.1 every runnable cell is stable and none is thereby valid. Split-half reliability thus carries no validity signal in either direction. Protocol and complete tables: Appendix A.19 (Tables 11, 12 and 13).

Table 2: **The landscape matrix** (certified shape; headroom H in pp). Ungradable cells ($z < 1$) grade none. Full statistics: Tables 4 and 10.

	Code	JSON Schema	Regex	QA
Gemma-2 9B	flat (33.3)	late peak (64.8)	flat (84.7)	ungr. (2.7)
Qwen-2.5 7B	peak (27.3)	unresolved (9.6)	late plateau (57.3)	ungr. (1.1)
Llama-3.1 8B	random wins (29.0)	early cliff (49.2)	early peak (93.0)	ungr. (3.8)

Detection is a commodity. If placement fails, does the SAE signal at least *detect* the shift? It does—but not distinctively: it never beats the best simple baseline, and the best detector is domain-dependent (Appendix A.5). Detecting drift is easy; *where to adapt* is the hard, unsolved part.

4.7 Boundary conditions: which domains can grade localizers

Table 10 reports $H/z/R$ for *every* attempted domain×model cell, including those never swept. Two failures bound it: an induction-pattern task arrived solved (≥ 0.95 everywhere); a traditional-Chinese conversion task died the opposite way—no shift (base $\approx T_1 \approx T_2$). A pinyin-transliteration domain sits between: adaptation works, headroom is unresolvable at this n ($H = 4.8$pp, $z = 1.1$, $R = 1.6$). The regex dialect grades admissible everywhere ($H = 84.7$pp, $z = 39.0$, $R = 53.1$ on Gemma-2; $H = 93.0$pp, $R = 85.7$ on Llama-3.1) and defeats the battery (§4.2). Constructing a domain that is unsolved, actually shifts, and yields resolvable headroom is nontrivial; the admissibility statistics ship with the benchmark.

QA: learned but not decodable. Why is QA's headroom within noise? The facts *do* enter the weights: across the placement grid, gold-answer negative log-likelihood drops substantially (mid-layer windows reach $\Delta\text{NLL} \approx -0.6$ to -0.9) while greedy held-out accuracy stays flat—the model absorbs the T_2 distribution but cannot decode it into correct greedy answers under this budget. NLL is therefore placement-sensitive but accuracy-irrelevant, reported descriptively, never as a grading trigger: representational movement and decodable behavior can dissociate (Figure 3; Appendix A.20)—which is why localizer benchmarks must gate on the behavioral metric they predict. The not-decodable verdict replicates on Qwen-2.5 7B ($H = 1.1$pp) and Llama-3.1 8B ($H = 3.8$pp, $z = 0.8$; Table 10), and so does the dissociation: every arm of Llama-3.1's ten-arm QA grid moves gold-answer NLL (all negative, mean -0.40, near-monotone in depth) while accuracy stays flat.

5 Discussion

What a negative benchmark establishes. We do not claim interpretability localizers are useless in general, only that none of seven reliably predicts placement against a ground-truth grid. The value is a calibrated ceiling: future localizers score against the same grid, and "we localized the right layers" becomes falsifiable rather than illustrative.

Cross-architecture generality. To test whether these results are Gemma artifacts, we ran the same grid, windows remapped by depth fraction, on Qwen-2.5 7B (code, JSON, regex) and Llama-3.1 8B (all four domains; an earlier Llama-3 8B code grid, the depth-band anchor, is a checkpoint contrast). The matrix's verdict column is uniform: in no cell does any localizer both hit and beat random (Table 4, Appendix A.11; Table 10).

Four consequences follow (Table 2). First, landscape *shape* is a joint property of base model and domain: all three replicated domains change shape across architectures—and on Llama-3.1 the verdict is domain-contingent *within* one architecture (whole-network beats the ten-arm field under Bonferroni on code and regex, $10/10$, but only $5/10$ on JSON). Second, the depth-fraction hypothesis fails out-of-sample: the first two architectures' resolved optima cluster at 57–61% relative depth, but the intersection band $[0.554,\ 0.595]$, frozen with its grading rule before any later landscape was unblinded (Appendix A.16), breaks on every graded Llama cell—four breaks across two checkpoints and three domains. A depth fraction that holds on two architectures is not a placement rule. Third, no Llama-3.1 domain's best window beats its best random draw (code: the draw wins outright, $p = 0.0018$; regex: exact tie, $p = 1.0$; JSON: unresolved null, $p = 0.375$). Fourth, headroom is

necessary but not sufficient for structure: Qwen-JSON's compressed headroom suppresses detectable structure, yet at similar headroom Gemma-code is statistically flat, Qwen-code peaks, and Llama-3.1 code resolves real structure whose top loses to random.

No single toolchain covers any model (no usable SAE for Qwen-2.5, no tuned lens for Gemma); on Llama, coverage is one generation (Llama Scope targets 3.1, not 3) and one SAE width (32×; 8× fails construct validity in all four domains). The four portable localizers (Fisher, gradient-direction, probe, AdaLoRA) ran on both resolved Qwen landscapes, Llama-3 code, and all four Llama-3.1 domains, plus the Llama Scope 32× SAE (Qwen-JSON unresolvable; descriptive only). The negative is architecturally robust: the two Llama-3 nominal hits are the same boundary-snapped arm, and the Gemma-topping depth heuristic misses all three replication code grids. On Llama-3.1 the failure sharpens: three of the four portable localizers return landscape-invariant readouts in every domain, so their nominal hits on early-peaked landscapes are coincidence, not task sensitivity; AdaLoRA, the only domain-sensitive prediction, moves *against* the landscape, naming a late window on early-peaked JSON at $\approx 7\times$ mean random-draw regret (Appendix A.10).

What the localizers measure: profile–benefit concordance. Each full per-layer profile is scored for rank concordance with the benefit curve (per-window held-out accuracy; Kendall τ-b, exact $n{=}7$ permutation null, Stouffer-pooled across the four resolved landscapes; Table 14, Appendix A.21). Neither gradient localizer's profile is detectably concordant with benefit (Fisher pooled $Z{=}{+}1.58$, $p{=}0.11$; gradient-direction $Z{=}{+}0.91$, $p{=}0.36$; a low-power test: no detectable relationship claimed, not orthogonality). These instruments report where representations move—and that is not detectably where added capacity helps.

The discarded signal transfers: a cross-cell readout. Is the failure in the profiles, or in the per-cell rules that reduce them? Every rule collapses a per-layer profile to a single argmax, discarding its shape. As a test of what the reduction destroys, an eighth localizer keeps the shape and borrows strength across cells: a ridge readout over the four window-mean localizer profiles plus window-center depth and depth^2, fit on the other eight graded cells, names each held-out cell's predicted-benefit argmax—a fixed rule calibrated on *other* cells, never the graded cell's training. Under the identical battery it identifies a top-2 window in $8/9$ cells (depth band as predictor $4/9$; fixed depth fractions $4/9$; AdaLoRA's rule $3/9$; chance $\geq 2.6/9$; binomial $p{\approx}3{\times}10^{-4}$, approximate) and on Qwen-regex clears the corrected certification bar (Appendix A.10's caveats travel). The transferable signal is almost entirely AdaLoRA's retained-rank profile—the one profile computed from adaptation dynamics rather than a static readout—the signal every per-cell decision rule discards. Its form was chosen from a pooled analysis of the same grids and consumes eight cells as calibration: a use case for the released benchmark, not a rescue of per-cell localization (Appendix A.22).

Limitations. A different budget or rank could reshape the grid (both fixed); AdaLoRA enters above the 112-rank budget class—a logged confound, within-cell grading unaffected (Appendix A.4). Placement is graded as *contiguous* windows plus the uniform arms—deployed practice (Hu et al., 2022; Dettmers et al., 2023); scattered-set placement is a distinct benchmark. The benchmark is greedy-decoding, held-out-accuracy based, and all grids are 7–9B; we claim nothing at frontier scale—the admissibility statistics gate any larger grid, and three families over two checkpoint generations rule out single-model idiosyncrasy. Finally, "no localizer works" is bounded by our seven; the released grid admits more.

6 Conclusion

We reframed capability-drift adaptation as a benchmarkable question—which layers should receive the update. Across seven localizers, four domains, and three model families—twelve cells—none reliably predicts placement against the ground-truth grid: where flat, random ties everything; where peaked, even naming the optimum fails to beat a random matched-budget window under corrected tests. PlacementBench, not any localizer, is the contribution. Its graded cells double as calibration data: a cross-cell readout of the discarded profiles identifies a top-2 window in $8/9$ held-out cells and clears the certification bar on one—hitting and certifying on the same cell, which no per-cell localizer did (Appendix A.22). Code and data: `https://anonymous.4open.science/r/PlacementBench-FE7C`

REPRODUCIBILITY STATEMENT

All models (Gemma-2, Qwen-2.5, Llama-3, Llama-3.1) and SAEs (Gemma Scope, Llama Scope) are public. The placement grid, localizer battery, scoring rule, and admissibility criteria are released as PlacementBench with their frozen rules and freeze timestamps. All graded runs are single-GPU by protocol (multi-GPU data parallelism silently scales the effective batch and step count; the released runners hard-stop if more than one device is visible), and fixed-seed retraining reproduces per-item vectors only to within 1–4 items of 250 on this stack, so repro checks are tolerance-based. An anonymized artifact—per-item correctness vectors, frozen scoring and reduction rules, timestamped contrast families, and the eligibility audit—accompanies the submission.

AI USE STATEMENT

LLMs appear in this work in three distinct roles. **As the subjects of study** (documented in the body): Gemma-2 9B, Qwen-2.5 7B, and Llama-3 / Llama-3.1 8B are the models adapted and graded on every placement grid, and the SAE localizers read publicly released sparse autoencoders trained on these models (Gemma Scope, Llama Scope). No LLM acts as a judge or simulator anywhere in the evaluation pipeline: grading is deterministic—greedy decoding with exact-match or substring scoring against held-out references (Appendix A.8). **As a data-construction aid**: benchmark construction was LLM-assisted. The code, JSON-Schema, and regex shifts are anchored to external specifications (PEP 695/701, JSON-Schema drafts, the ECMAScript regex dialect) and graded by exact match on spec-defined constructs, so no graded result depends on the factual accuracy of LLM-drafted text. The factual-QA pairs are LLM-drafted against an author-designed 2023→2024 knowledge-cutoff schema (the categories and the detection-versus-production design are the authors'); QA grades uninformative and enters no placement verdict (§3.4), so pair-level factual accuracy affects no scored result. **As a writing aid**, the authors used an LLM to assist with drafting and editing the manuscript prose; all technical content, experimental design, analysis, and claims are the authors' own, and the authors take full responsibility for the paper's content.

REFERENCES

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016.

Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022.

Nora Belrose, Igor Ostrovsky, Lev McKinney, Zach Furman, Logan Smith, Danny Halawi, Stella Biderman, and Jacob Steinhardt. Eliciting latent predictions from transformers with the tuned lens. *arXiv preprint arXiv:2303.08112*, 2023.

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. *Transformer Circuits Thread*, 2023.

Arthur Conmy, Augustine N Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards automated circuit discovery for mechanistic interpretability. In *Advances in Neural Information Processing Systems*, 2023.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600*, 2023.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. QLoRA: Efficient finetuning of quantized LLMs. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Ning Ding, Xingtai Lv, Qiaosen Wang, Yulin Chen, Bowen Zhou, Zhiyuan Liu, and Maosong Sun. Sparse low-rank adaptation of pre-trained language models. *arXiv preprint arXiv:2311.11696*, 2023. EMNLP 2023.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, et al. Toy models of superposition. *Transformer Circuits Thread*, 2022.

Gemma Team. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. Dissecting recall of factual associations in auto-regressive language models. *arXiv preprint arXiv:2304.14767*, 2023.

Peter Hase, Mohit Bansal, Been Kim, and Asma Ghandeharioun. Does localization inform editing? surprising differences in causality-based localization vs. knowledge editing in language models. In *Advances in Neural Information Processing Systems*, 2023.

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. *arXiv preprint arXiv:2106.09685*, 2022. ICLR 2022.

Anand Kamat, Daniel Blake, and Brent M. Werness. Grad detect: Gradient-based hallucination detection in LLMs. *arXiv preprint arXiv:2606.24790*, 2026. 2nd Workshop on Compositional Learning: Safety, Interpretability, and Agents, ICML 2026 (non-archival).

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114 (13):3521–3526, 2017.

János Kramár, Tom Lieberum, Rohin Shah, and Neel Nanda. Atp*: An efficient and scalable method for localizing LLM behaviour to components. *arXiv preprint arXiv:2403.00745*, 2024.

Yoonho Lee, Annie S. Chen, Fahim Tajwar, Ananya Kumar, Huaxiu Yao, Percy Liang, and Chelsea Finn. Surgical fine-tuning improves adaptation to distribution shifts. In *International Conference on Learning Representations*, 2023.

Raymond Li, Loubna Ben Allal, Yangtian Zi, Niklas Muennighoff, Denis Kocetkov, Chenghao Mou, Marc Marone, Christopher Akiki, Jia Li, Jenny Chim, et al. Starcoder: may the source be with you! *arXiv preprint arXiv:2305.06161*, 2023.

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.

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.

Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, 1947.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, 2022.

Rui Pan, Xiang Liu, Shizhe Diao, Renjie Pi, Jipeng Zhang, Chi Han, and Tong Zhang. LISA: Layer-wise importance sampling for memory-efficient large language model fine-tuning. In *Advances in Neural Information Processing Systems*, 2024.

Xu Wang, Yan Hu, Wenyu Du, Reynold Cheng, Benyou Wang, and Difan Zou. Towards understanding fine-tuning mechanisms of LLMs via circuit analysis. *arXiv preprint arXiv:2502.11812*, 2025.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Nikos Karampatziakis, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. Adalora: Adaptive budget allocation for parameter-efficient fine-tuning. In *International Conference on Learning Representations*, 2023.

A Appendix

A.1 Vocabulary and Notation

Terms used throughout, defined once. Each is scoped exactly as used in the text.

No-T_2 floor	Held-out accuracy of the T_1-adapted model before any T_2 adaptation; the baseline every arm is measured against.
Headroom H	Whole-network adaptation accuracy minus the no-T_2 floor: the largest accuracy movement the domain offers any placement.
z	H in units of an unpaired difference-of-proportions standard error; asks whether any headroom exists at all.
Resolvability R	H in units of a single sweep point's binomial error bar; an upper bound on how many distinguishable accuracy levels the landscape could contain. z and R are distinct statistics and are never interchanged.
Matched-budget	Trained at the grid's fixed LoRA parameter budget, so arms differ only in placement, never in capacity.
Window names	Named arms encode their literal layer span on the Gemma-2 grid: *detection* = layers 10–18 (c14), *boundary* = layers 18–28, *downstream* = layers 20–28 (c24) and 22–30 (c26). The prefixes are descriptive holdovers from the SAE analysis (the early band where shifted features first appear vs. later processing bands); they mark no construction or grading difference from any other matched-budget arm.
Random draw	A random matched-budget placement, graded identically to every window arm; the frozen draws are what every advantage claim is tested against.
Regret	The held-out accuracy gap between the window a localizer names and the realized optimum of the same grid.
Designated null	A contrast for which non-significance is the wanted outcome (e.g., a named window vs. a random draw), fixed as a null before testing.
Identification	Naming a top-2 window of the realized landscape (tie-inclusive).
Certified advantage	The named window beats the matched-budget random draws under paired exact McNemar with family-wise Bonferroni correction.
Hook-anchored	The localizer reads a single residual-stream hook and its named window cannot exceed that layer; *hook-capped*: the landscape's peak lies above the cap.
DIFFUSE	Abstention grade: the profile's $\geq 50\%$-of-max band spans more than half the network, so no window is named.
Boundary snap	A prediction whose raw center falls outside the tested grid clamps to the nearest grid arm; reported as a clamp, not as a prediction.
Selection-inflated	The realized optimum is the maximum of the same sweep it is tested on; its edge over draws carries a winner's-curse bias and is never credited to any localizer.
Depth band	The intersection of top-2 depth-fraction regions from the first two architectures, frozen with its grading rule before the third architecture's grid was unblinded.

A.2 Localizer Specifications

Each localizer is frozen by a specification document written before grading; we summarize the load-bearing decisions.

A.2.1 AtP+GradDrop

We adapt AtP* (Kramár et al., 2024) to per-layer residual-stream nodes with a mean-ablation reference. Of AtP*'s two corrections we implement exactly one: **GradDrop** (Eq. 11 of Kramár et al. 2024) ports verbatim as a per-dropped-layer gradient-zeroing aggregated as $\hat{c}(n) = \mathbb{E}[(1/(L-1))\sum_{\ell} |\hat{I}_{\text{AtP+GD}_\ell}(n)|]$ with $L =$ layers upstream of the SAE hook. The **query/key saturation fix** is structurally inapplicable to layer-granularity nodes (it is defined per-head on the softmax

input, and correcting it at layer scale reduces to full activation patching), so we document it out and label the method "AtP+GradDrop," not AtP*. We treat the LayerNorm denominator as stop-gradient per Kramár et al. (2024) §5.2, appropriate for coarse residual nodes.

A.2.2 LINEAR PROBE

The emergence-layer statistic (first layer $\geq 90\%$ of cross-layer max held-out T_1/T_2 discrimination accuracy) is frozen over raw accuracy to avoid the separable-everywhere ceiling; a per-layer standardizer fit on train only and a fixed regularization constant keep the probe knob-free (relevant to the fairness distinction that admits the probe but excludes a tuned lens). A domain grades diffuse if max accuracy is below a headroom-scaled threshold.

A.2.3 SCORING RULE

The band→window map (§3.3) is shared verbatim across all profile-based localizers via a single implementation, so no tie-breaking or grid-snapping difference can be introduced per method. Per-localizer snapping in Table 1: Fisher's raw center c25 snaps to the c26 window (displayed as scored, matching Figure 1); SAE-16k's JSON c16 (displayed raw, between grid windows) is scored as c18; gradient-direction's code c38 (displayed raw, beyond the last grid window) is scored as c34; AdaLoRA's code allocation peaks at L39, beyond the last grid window, and snaps to the edge window c34.

Two scoring notes. First, we score under the tie-inclusive top-2 criterion, the reading that *helps* the localizers, and also report the stricter exact-top-2 verdict, under which Fisher's scored c26 window grades a miss, the battery goes $0/7$ on code, and AdaLoRA's c30 (the optimum, a hit under any criterion) is the sole JSON survivor. The per-item tests support treating c24 and c26 as one tied level: the two windows (both 0.812) agree on $246/250$ items (discordance $4/250$; McNemar $p = 1$ *by construction* when aggregates are equal, so the discordance count is the evidence)—a genuine same-items tie, not two windows succeeding on different items with coincidentally equal totals. Second, the two natural reductions of a profile to a window (uniform band-center vs. the pre-frozen band skeleton) disagree in 3 of 12 cells among the six untrained localizers (Fisher-code: DIFFUSE vs. c34; SAE-131k-code: c18 vs. c22; Fisher-JSON: c26 vs. c30, both hits). We score with the single uniform rule throughout; the disagreements are themselves an instrument-sensitivity finding—the same profile can name windows one to two grid steps apart depending on an innocuous-looking reduction choice.

A.3 ALL-LAYER SAE HOOK SWEEP (JSON)

Detail for the sweep of §4.3. All three hook-anchored localizers are re-extracted at every residual-stream hook 0–41 (Gemma Scope provides canonical SAEs at both widths for all 42 layers); feature selection and attribution are re-run at each hook—AtP+GradDrop re-runs its full per-layer drop battery, its five mechanical asserts passing at every hook—and the scoring rule of §3.3 is unchanged. Answer key: tie-inclusive top-2 windows {c24, c26, c30}; optimum c30 (0.840). Two hook-0 cells are structurally N/A: SAE-131k fails a pre-specified degeneracy guard (its attribution profile concentrates entirely on layer 0), and GradDrop's normalizer $1/(L-1)$ is undefined at $L{=}1$ (a designed no-file exit); the remaining 124 of 126 cells grade cleanly (Table 3).

The sweep sharpens the two-width diagnostic: SAE-131k's Table 1 miss is attributable to hook reach—given a hook at or past the peak it names c30—while SAE-16k's failure is a resolution limit, not a reach limit: at no hook depth does the 16k width name c30 (its best case is the $+0.028$ tie at hooks 27–31; at hook 34 its top-5 attribution set is bimodal, $\{16, 25, 26, 29, 30\}$, and the diffuseness guard fires). At hook 34 the 131k profile's skeleton reduction names c34, a miss—the uniform-vs-skeleton disagreement of §4.2 recurs; verdicts everywhere use the single uniform rule.

AtP+GradDrop is the sweep's sharpest negative. The computation is hook-sensitive—the profile spans two layers at hook 1 and all 42 at hook 41, with per-hook feature sets to match—but the verdict is invariant: the mapped center lands at c3–c13, below the shallowest tested window, at every one of the 41 gradable hooks. Hooks 21–41 pin the center at c4–c5 (at hook 41, layer 0 carries $2.4\times$ the mass of any other layer); mid-depth truncation lets it drift as high as c13 (hooks 6–12, 19) without ever reaching the grid. The graded c14 is therefore a grid clamp (the extrapolation flag is set in every

Table 3: All-layer hook sweep, JSON domain: SAE verdicts under the single frozen uniform rule as a function of hook depth, grouped by verdict regime. Regret = optimum accuracy minus the predicted window's accuracy. The hook-20 row reproduces Table 1 exactly (16k raw center c16 snaps to c18); the hook-34/36 SAE-131k cells name the optimum and inherit its raw-only edge over the best random draw ($p = 0.019$, fails Bonferroni; §4.4); the hook-27–31 hits name the c24/c26 tie, which, like Fisher's window, was not among the frozen contrasts and is left ungraded against random. AtP+GradDrop is hook-invariant and shown as a single row: its predicted center falls below the tested grid at every gradable hook and clamps to the shallowest window.

Hook(s)	SAE-16k	SAE-131k
0	c0 × miss (+0.120)	N/A (degenerate)
1–18	× miss (+0.120)	× miss (+0.120)
19	× miss (+0.096)	× miss (+0.096)
20 (frozen)	c18 × miss (+0.096)	c18 × miss (+0.096)
21–26	× miss (+0.068 to +0.096)	× miss (+0.068 to +0.096)
27–31	c24/c26 ✓ hit (+0.028)	c24/c26 ✓ hit (+0.028)
32–33, 35	abstain	abstain
34, 36	abstain	c30 ✓ hit (= opt., +0.000)
37, 39	abstain	× miss (+0.036)
38	× miss (+0.036)	abstain
40–41	× miss (+0.036)	× miss (+0.036)
AtP+GradDrop, hooks 1–41: c3–c13 (below grid) → c14 (extrap.) × miss (+0.120); hook 0 N/A		

cell), not AtP's prediction: its signal points below the entire tested grid at every hook. The sweep is JSON-only: code's plateau offers no late target, QA is ungraded, and no SAE toolchain exists for Qwen-2.5 at any layer (§5).

A.4 AdaLoRA as a Trained Localizer

Protocol for the battery's seventh localizer (§3.2). A fresh AdaLoRA adapter (q/k/v/o, every layer; init rank 12, target average rank 1) is trained from the same base+T_1 starting function the grid arms adapt from: the T_1 LoRA is merged into a bf16 base, and merge fidelity is certified empirically on every cell (base+T_1 accuracy on a 50-item T_1 slice before vs. after merge; $|\Delta| = 0$ items on all thirteen cells). Training data, epochs, sequence length, and learning rate match the cell's grid arms exactly; the allocator's budget schedule is calibrated to the true optimizer-step count and its update is invoked at every step (verified by exact call-count asserts, and guarded by a hard failure if retained rank ever collapses to zero—a silent failure mode a plain Hugging Face `Trainer` produces by never invoking the allocator's update). The placement signal is the final per-layer retained rank (nonzero singular triplets after pruning), reduced to a window by the same frozen rule that scores every profile localizer (§3.3). The minimum feasible all-layer budget (target average rank 1) differs by architecture—$168/112/128$ retained rank on Gemma/Qwen/Llama versus the grid's 112 window class—a logged cross-architecture confound; within-cell grading is unaffected. The Gemma base is bf16 here versus the grid's NF4 (a lossless merge requires an unquantized base); the bf16 quantization-control tier shows the Gemma landscape ties persist at full precision, and this defense extends to Qwen/Llama by assumption rather than by a dedicated control.

All thirteen cells: Gemma code c34 (miss, regret $+0.067$), JSON **c30 (hit = optimum, regret** 0), regex c34 (miss, $+0.033$), QA descriptive (c24); Qwen code c9 (miss, $+0.173$), JSON c23 (descriptive hit on an unresolvable landscape), regex c23 (miss, $+0.050$), QA descriptive (c23); Llama-3 code c11 (nominal hit on the boundary arm; paired McNemar vs. best random $p = 0.080$, point estimate below it); Llama-3.1 code c11 (nominal hit, regret $+0.007$, on the grid where the best draw beats every window), JSON c23 (miss: a late window on the early-peaked cliff, $\approx 7\times$ the mean random-draw regret), regex c17 (miss, regret $+0.033$ vs. the random mean's $+0.029$; the raw allocation peak, L14, is itself a top-2 center—the frozen band-to-window reduction produced the miss, and both facts are reported), QA descriptive (c11). Across the four Llama-3.1 cells AdaLoRA is the only localizer whose prediction moves, and its movement is anti-correlated with the landscape. AdaLoRA's own held-out accuracy at this matched budget is low in most cells (0.06–0.38; the JSON cells reach 0.80 on Qwen and 0.51 on Llama-3.1): the allocation is graded regardless of

fit level, exactly as the battery's untrained localizers are graded without fitting anything—and the near-floor fit at the grid's budget class is itself a matched-budget property of the method. Descriptively, the allocation reproduces the battery's edge artifacts at both grid edges (late edge c34 on Gemma code/regex, early edge c11 on Llama-3), and on identical code data its profile peaks at $\sim$93%, 11%, and 3% of depth on Gemma, Qwen, and Llama-3 respectively—the same task, opposite ends of the network by architecture.

A.5 Detection Battery (context)

Full detection results (SAE feature drift vs. perplexity and other baselines, per domain and contamination level) supporting §4.6. SAE drift correlates with degradation but does not exceed the best baseline in any setting; the low-contamination regime is uninformative for all detectors. The best baseline is domain-dependent: perplexity is strongest for code and QA but blind to JSON schema drift, which moves representations without moving next-token likelihood.

A.6 Feature-Level Diagnosis (context)

When SAE features are inspected, top "birthed" features (high on T_2, near-zero on T_1) are often monosemantic for the shifted construct—e.g. PEP 695 type parameter syntax in the code domain, `prefixItems`/`$dynamicRef` in JSON Schema. We report this as qualitative context: diagnostic specificity is real but is not what the placement benchmark tests, and it does not translate into a correct *placement* prediction (§4).

A.7 Gradient Distribution

Fisher and gradient-direction profiles are broadly distributed across layers (normalized entropy > 0.97 for gradient magnitude), consistent with the gradient-selected window tying random on the code grid. This contrasts with Kamat et al. (2026), who find hallucination-relevant gradient signal concentrated in the final five layers — a different task and label.

A.8 Evaluation Protocol

Greedy decoding (`do_sample=False`); exact-match on target constructs for code/schema, substring containment for QA. T_2 split into train / held-out before experimentation; all accuracies on held-out data.

A.9 Fine-Tuning and Grid Construction

T_1: base + LoRA (rank 16, $\alpha = 32$) on all projections, 3 epochs. Grid arms: load base + T_1 adapter (active, not merged), freeze all, selectively unfreeze LoRA parameters in the arm's target window, and train on T_2 under identical hyperparameters, holding trainable-parameter count fixed within each budget class. T_2 epochs are per-domain, held constant across all arms and all architectures within a domain, so every within-domain comparison is at matched training budget. Budgets were set to yield gradable landscapes: JSON Schema at 1 epoch (at 3, every arm saturates within 2.4pp of a 0.980 ceiling, collapsing the landscape); code at 3 epochs (at 1, arms floor and compress toward the no-T_2 baseline—the ordering direction holds but is not resolved by the available unpaired test, $z = 1.84$); regex dialect at 3 epochs (measured gradable at preflight: unsaturated, $H = 84.7$pp). QA uses 3 epochs, consistent with the other multi-epoch domains. A capacity-doubled tier retrains T_1 and the full code-domain family at rank 32 ($\alpha = 64$, matched α/r) in both architectures (§4.5); ranks are distinct budget classes and are never compared by paired test. The 9B model uses 4-bit NF4 quantization; single-GPU execution for consistent step counts (multi-GPU data parallelism silently scales the effective batch; every runner hard-stops if more than one device is visible). Full-precision (bf16) robustness checks on the code- and JSON-domain regret families confirm neither verdict is a quantization artifact (§4.5).

A.10 Cross-Architecture Replication Protocol

The Qwen-2.5 7B tier (28 layers, same 4-bit NF4 quantization and single-GPU regime) replicates the grid protocol, not the toolchain: the seven nine-layer Gemma windows are remapped by depth fraction onto 28 layers (centers 9–23), joined by three matched-budget random-7 draws and whole-network adaptation (eleven arms per domain). Within a domain, T_2 data, epochs, and trainable budget class are held identical to Gemma's; the matched-budget window class trains $2{,}523{,}136$ parameters at $r = 16$. Domains attempted on Qwen are exactly those that graded gradable on Gemma, plus QA as a cross-architecture admissibility check (Table 10). No pretrained SAE with the required layer coverage and quality exists for Qwen-2.5, which removes SAE-16k, SAE-131k, and AtP+GradDrop (its target features are SAE-derived) from the portable battery; this absence is reported as the portability finding, not imputed around.

The Llama-3 8B tier (32 layers, same 4-bit NF4 quantization and single-GPU regime) first extended the protocol to a third architecture for the code domain—the domain that anchors the depth band (Appendix A.16). The seven Gemma windows are remapped by depth fraction onto 32 layers (centers 11–26), joined by three matched-budget random-7 draws and whole-network adaptation (eleven arms); T_2 data, epochs, sequence length, and budget class are identical to the code-domain protocol on the other two architectures. No pretrained SAE with the required layer coverage exists for Llama-3 8B, so the battery is again the portable trio. T_1 retention for all eleven arms is evaluated on the full T_1 held-out split.

The Llama-3.1 8B tier (same 32-layer geometry, quantization, and single-GPU regime) is the study's full third architecture: all four domains under the identical arm design (seven remapped windows at native centers 11–26, three matched-budget random-7 draws with frozen seeds, whole-network adaptation), with T_2 data, epochs, sequence length, and budget class held to each domain's protocol on the other architectures; the earlier Llama-3 code grid is retained above as a checkpoint contrast. Unlike Qwen-2.5 and Llama-3, a pretrained SAE with the required coverage exists for this checkpoint (Llama Scope), so the battery extends beyond the portable trio: the $32\times$ width passes the frozen construct-validity gate at the analysis hook in every domain (FVU 0.459–0.470 against the < 0.5 bar, on-distribution L_0 in range), while the released $8\times$ width fails the same gate in every domain (FVU 0.558–0.564)—a width-level failure reported as a finding, not a caveat. SAE placement profiles are not reported for Llama-3.1: the placement construction taps Gemma-2's post-feedforward sandwich-norm, which has no Llama analog; availability and admissibility port, the placement construct does not.

A.10.1 Per-domain results

Full statistics for the cross-architecture summary in §5.

JSON Schema ($H = 9.6$pp, $z = 2.9$, $R = 4.6$; $n = 250$). The landscape is flat: the seven principled centers span 0.840–0.868 (a 2.8pp span, on the order of a single eval-noise unit), random draws reach 0.876, and Gemma-JSON's late peak does not reappear—the depth-fraction analog of Gemma's optimum (center 20, $\sim$71% depth) ties the plateau at 0.868. Per-item McNemar tests certify the tie: the depth-analog window and each random draw disagree on only 2–4 items of 250 (all $p \geq 0.5$)—same-item placement irrelevance, not churn with conserved aggregates. The only structure the domain resolves is that the earliest window is genuinely worse ($p = 0.0039$, Bonferroni $m = 5$, 0.010 threshold); the depth-analog window trails raw ($p = 0.039$) without correction.

Code ($H = 27.3$pp, $z = 7.6$, $R = 9.5$; $n = 300$). The landscape is peaked: the seven principled centers span 0.203–0.387 (18.4pp, $\approx$6.5 SEs), with a unimodal maximum at center 17 ($\sim$61% depth). Per-item McNemar tests of the frozen family confirm the structure. The depth-fraction analog of Gemma's code optimum (center 16, $\sim$57% depth) is indistinguishable from all three random draws (23–44 discordant items of 300; all $p \geq 0.40$)—with 8–15% item-level churn rather than the near-zero same-item ties of the JSON domain, confirming that mid-window placement changes *which* items are solved rather than how many. The early-window deficit is real: center 16 beats center 9 (earliest) on 48 vs. 9 discordant items ($p < 0.001$), and the best random draw beats center 9 similarly ($p < 0.001$); both survive Bonferroni correction ($m = 5$).

Regex dialect ($H = 57.3$pp, $z = 18.1$, $R = 36.6$; $n = 300$). The domain that resolves flat on Gemma-2 resolves *structured* on Qwen-2.5: a raised late plateau. The empirical maximum (center 20,

$\sim$71% depth, 0.833) beats all three random matched-budget draws (paired McNemar $p = 0.0002$, $< 10^{-4}$, 0.0020; all Bonferroni-significant, $m = 9$) and the earliest window ($p = 0.0001$), but is statistically indistinguishable from its neighboring windows (centers 12, 15, and 17; 0.800–0.803; three post-hoc neighbor contrasts, uncorrected $p = 0.049$–0.053, none Bonferroni-significant)—so the licensed reading is a late plateau that beats random, not a peak at center 20. Center 20 is moreover the realized maximum of this sweep—the same selection caveat applied to Gemma-JSON's optimum (§4.2)—and the full seven-arm family is re-tested under two additional training seeds (Table 7): the plateau's edges over two of the three draws and over the earliest window replicate at every seed, while its edge over the strongest draw is direction-consistent in $3/3$ seeds but Bonferroni-significant in $1/3$ ($p = 0.0020/0.146/0.0225$)—the selection-predicted softening. The certified-advantage reading rests on the plateau's Bonferroni-significant draw contrasts at the primary seed, not on the identity of the maximizing window. The depth-fraction analog of Gemma's flat-top (center 16, 0.797) separates from only the weakest of the three draws ($p = 0.057$, 0.0003, 1.00). Whole-network adaptation again beats every window (0.920 vs. 0.833, $p < 10^{-4}$). Against this structure the four portable localizers score $0/4$: Fisher's uniform reduction grades DIFFUSE, gradient-direction names c15 (regret $+0.033$), AdaLoRA names c23 (miss, regret $+0.050$, $\approx$ the random mean's $+0.064$), and the probe is saturation-degenerate as in every Qwen domain.

Third architecture: Llama-3 8B, code ($H = 32.3$pp, $z = 8.8$, $R = 11.2$; $n = 300$). Admissibility statistics are nearly identical to Gemma-code (33.3pp, 8.8, 11.7)—and the landscape takes a third shape. The grid resolves real structure: early windows beat late (center 11 vs. center 23, $p = 0.0001$; windowed maximum vs. center 23, $p < 10^{-4}$), inverting the early-window deficit of Qwen-code. Yet the windowed maximum (center 14, $\sim$44% depth, 0.397) is statistically indistinguishable from all three matched-budget random draws ($p = 1.00$, 1.00, 0.43; the best draw ties it exactly at 0.397 on 19+19 discordant items—churn with conserved aggregates), and whole-network adaptation beats the best window (0.513 vs. 0.397, $p < 10^{-4}$). The portable battery does not localize: Fisher's band again collapses to the network boundary (layers 1–2, as at layer 1 on Qwen-code), with its raw center below the grid; the floor-snapped earliest arm happens to sit in the top-2 tie (a nominal hit, regret $+0.037$) yet confers less than the random-draw mean (regret $+0.008$; best draw 0.000), so the hit is a boundary snap with no advantage over random placement. AdaLoRA independently reproduces the same artifact: its allocation peaks at L1 ($\sim$3% depth) and snaps to the same earliest window—a second nominal hit on the identical boundary arm, which a paired test confirms does not beat the best random draw (exact McNemar $p = 0.080$, point estimate 0.360 vs. 0.397, i.e. below it). Gradient-direction names a late center (miss, regret $+0.090$), the probe is saturation-degenerate, and all four depth fractions miss (regret $+0.067$ to $+0.140$). Descriptively, whole-network adaptation also pays the grid's worst T_1 retention (0.780 vs. 0.860–0.995 for every windowed and random arm): the full-budget ceiling is bought with forgetting that every placement avoids.

Third architecture, full grid: Llama-3.1 8B, code ($H = 29.0$pp, $z = 7.7$, $R = 10.0$; $n = 300$). Whole-network adaptation (0.513) beats the entire ten-arm field under Bonferroni ($10/10$ at $m = 10$). The window field's top is a near-tie between the earliest and latest centers (centers 11/14 at 0.310, center 26 at 0.317) while the in-band anchor (center 18) scores 0.240—and the decisive contrast is the random one: the best matched-budget draw (0.397) beats the best window (0.317; $n_{10}/n_{01} = 16/40$, $p = 0.0018$; both arms post-hoc maxima, a framing least favorable to this reading's rejection). Random placement does not merely tie the windows on this grid; it beats them.

Llama-3.1 8B, JSON Schema ($H = 49.2$pp, $z = 13.4$, $R = 25.1$; $n = 250$). The landscape is an early-peaked monotone cliff: the early centers score 0.880 (centers 11/14) and decline to 0.648 at center 26. Here whole-network adaptation (0.892) is *not* certified over the field: it separates from the four late windows ($p \approx 0$, 27–61 discordant items) and one draw ($5/10$ under Bonferroni) but is statistically indistinguishable from the three early windows—so the landscape verdict is domain-contingent *within* a single architecture, where all prior divergence was cross-architecture. The best-window-vs-best-random contrast is an unresolved null ($p = 0.375$), and the in-band anchor (center 18) falls on the wrong side of the cliff—the band's third break.

Llama-3.1 8B, regex dialect ($H = 93.0$pp, $z = 62.0$, $R = 85.7$; $n = 300$; the study's largest headroom). The cleanest landscape in the study: a strict monotone early peak (0.903 at center 11 declining to 0.750 at center 26), with whole-network adaptation (0.963) again beating the entire field ($10/10$). The strongest random draw (0.903) ties the best window *exactly* (nine discordant items each way, $p = 1.0$), and the in-band anchor loses to the out-of-band maximum ($n_{10}/n_{01} = 5/20$,

$p = 0.0041$)—the band's fourth break. Placement structure this landscape resolves emphatically; localizer or band advantage over random it resolves nowhere.

Llama-3.1 8B, factual QA ($H = 3.8$pp, $z = 0.8$, $R = 1.2$; $n = 183$). Ungradable, as on both other architectures; the full ten-arm grid was still run for symmetry and is flat (0.235–0.284, with a random draw the nominal maximum). Gold-answer NLL nevertheless moves at every arm (all ten negative, mean -0.40, near-monotone in depth)—the third replication of the QA dissociation, reported descriptively in Appendix A.20. Localizer outputs on this cell are recorded, never graded.

The Llama-3.1 battery. Three of the four portable localizers return the identical readout in all four domains: Fisher's band collapses to layers 1–3 and snaps to the earliest window (center 11) every time; gradient-direction is a diffuse abstention every time (on code its band spans layers 1–31); the linear probe is saturation-degenerate in code, JSON, and regex ($a_{\max} = 1.000$, emergence layer 1) and, on QA, deviates in its raw readout ($a_{\max} = 0.938$, emergence layer 3) yet still snaps to the same extrapolated earliest window—the raw readout varies, the prediction never does. Fisher's frozen center is a nominal top-2 hit on all three resolved landscapes, but a readout that never moves cannot be task-sensitive: the hits are where the landscapes happen to put their early peaks, and by the best-window-vs-random contrasts above, naming the best window confers no advantage over random placement on this architecture anyway. AdaLoRA is the only localizer whose prediction moves across domains—and it moves against the landscape: center 11 on code (nominal hit, regret $+0.007$), center 23 on JSON (a late window on the early-peaked cliff, $\approx 7\times$ the mean random-draw regret), center 17 on regex (miss, regret $+0.033$ vs. the random mean's $+0.029$; its raw allocation peak, L14, is itself a top-2 center—the frozen band-to-window reduction produced the miss, and both facts are reported). On code, the fixed depth fractions at $50/67/75\%$ all miss (regret $+0.020$–$+0.097$); only the untuned 90% fraction lands the realized maximum. The SAE tier is width-split as above: $32\times$ in-regime in all four domains, $8\times$ out-of-regime in all four; AtP+GradDrop is not ported to this architecture.

We ran the four portable localizers (Fisher, gradient-direction, linear probe, AdaLoRA) on both resolved Qwen landscapes (code and regex; the regex verdicts appear in its block above) and on the Llama-3 code landscape (verdicts likewise in its block above); the Qwen-JSON landscape is unresolvable ($R = 4.6$) and its localizer outputs are recorded as descriptive data only. On code, *zero of four* hit the {c16, c17} top-2. Fisher's band collapses to layer 1 (raw center 3, below the Qwen grid; the floor-snapped window is c9, the earliest arm and second-worst on the grid at 0.213; only c12, 0.203, scores lower). AdaLoRA's allocation peaks at layer 3 and names the same c9 window (miss, regret $+0.173$, worse than the random mean's $+0.051$)—the trained allocator converging on the untrained gradient localizer's floor-snap. Gradient-direction names c23 (×, regret $+0.060$). The linear probe is degenerate in all Qwen domains ($a_{\max} = 1.00$, emergence layer 1, UNINF-SAT). The depth-fraction heuristic that tops the Gemma-2 battery also misses here: 67–$75\% \times 28 = 18.8$–21 layers both snap to c20, which is outside the {c16, c17} top-2 (regret $+0.060$); all four fractions miss. On Llama-3 code all four fractions miss again (snapped centers 17–26, regret $+0.067$ to $+0.140$). On Qwen-regex the same c20 snap lands on the empirical maximum—a regret-0 landing, but on a plateau where that window is indistinguishable from its neighbors, so it is not evidence of a transferable rule where the code miss is evidence against one. Two distinct depth statements are in play and should not be conflated. The 67–75% *heuristic* is a fixed prescription that happens to land inside the top-2 regions of both Gemma landscapes; the 57–61% figure above is the *empirical* region where the resolved code optima cluster (on Gemma it also falls inside JSON's c24 = c26 = c30 top-2 tie). On Qwen-code the two come apart: 67–75% of 28 layers snaps to c20 and misses, while the empirical optimum sits at 61%. The heuristic's Gemma success thus reflects where its band happens to fall on that architecture, not a transferable rule—and no localizer in the battery identified the 57–61% region either. The negative is architecturally robust: on every resolved domain where portable localizers ran (Gemma-2 code, JSON, and regex; Qwen-2.5 code and regex; Llama-3 code; Llama-3.1 code, JSON, and regex), no portable localizer both hits and beats random—the two Llama-3 hits (Fisher and AdaLoRA) are the same boundary-snapped arm, which confers no advantage over random placement, and on Llama-3.1 no window of any kind beats the best random draw—and the depth shortcut does not transfer as a rule to any replication grid.

A.11 The Localizer Matrix

Table 4 assembles the study's full grid: three architectures × four domains, every cell run end-to-end. The final column is the benchmark's verdict, and it is uniform. The earlier Llama-3 8B code grid

Table 4: **The localizer matrix: three architectures × four domains.** Headroom statistics as in §3.4 (computed from unrounded proportions); the landscape column gives the certified reading of each cell (paired exact McNemar, per-domain Bonferroni); the last column asks whether *any* localizer in the cell's battery both named a top-2 window and beat a random matched-budget draw under paired, corrected tests. Ungradable cells ($z < 1$) grade no localizer; their outputs are recorded descriptively. Llama-3.1 window centers are native (11–26).

Model	Domain	n	H(pp)	z	R	Landscape (certified reading)	Hit ∧ beat random
Gemma-2 9B	Code	300	33.3	8.8	11.7	flat top: optimum/random/localizer 3-way tie	none $(0/7)$
Gemma-2 9B	JSON	250	64.8	19.6	36.2	late peak (c30); optimum > 2/3 draws	none $(0/7)$
Gemma-2 9B	Regex	300	84.7	39.0	53.1	resolved flat; max ties all draws	none $(0/7)$
Gemma-2 9B	QA	183	2.7	0.6	0.8	ungradable	—
Qwen-2.5 7B	Code	300	27.3	7.6	9.5	peaked (c17, ∼61% depth)	none $(0/4)$
Qwen-2.5 7B	JSON	250	9.6	2.9	4.6	unresolvable at this n	— (descriptive)
Qwen-2.5 7B	Regex	300	57.3	18.1	36.6	late plateau; beats draws under correction	none $(0/4)$
Qwen-2.5 7B	QA	183	1.1	0.3	0.4	ungradable	—
Llama-3.1 8B	Code	300	29.0	7.7	10.0	structured; best *draw* beats best window	none
Llama-3.1 8B	JSON	250	49.2	13.4	25.1	early cliff; uniform certified only $5/10$	none
Llama-3.1 8B	Regex	300	93.0	62.0	85.7	strict early peak; best draw ties max exactly	none
Llama-3.1 8B	QA	183	3.8	0.8	1.2	ungradable	—

(checkpoint contrast) behaves identically: its two nominal hits are one boundary-snapped arm that a paired test grades below the best random draw.

A.12 Matched-Cost All-Layer Control

A budget-matched *all-layer* arm separates placement from parameter count: the best window's total trainable budget is spread uniformly across every layer (rank 3–4 per layer), yielding an arm at window cost with whole-network coverage, run on all ten cells (both QA cells descriptive). The reading was fixed before unblinding: a win over the best window is a clean ceiling-raising result, while a loss is rank-starvation-confounded (per-layer rank 3–4 may sit below the floor at which any single adapter learns the shift) and licenses only "all-layer at this per-layer rank does not beat a window at matched budget," never "concentration wins." On the structured cells the arm fails to beat the best window in three of five (loses on Gemma-JSON, $p \approx 0$, and Qwen-regex, $p = 0.0075$; ties on Qwen-code)—but on Llama-3 code it **wins outright** ($p = 0.0003$) and ties the whole-network ceiling at $\sim$22% of its parameters (identical arm constructions on both generations of the same architecture). The Llama-3.1 code cell replicates this independently: the same construction (2,981,888 trainable parameters, equal to the window arms; rank 3–4 per layer) scores 0.477 against the best window's 0.317 ($n_{10}/n_{01} = 58/10$, $p = 2.4 \times 10^{-9}$) and is statistically indistinguishable from the whole-network ceiling (0.513; $21/32$, $p = 0.17$) at $\sim$22% of its parameters (2.98M vs. 13.6M)—an accuracy inside the ceiling's own training-seed spread (0.473–0.513). On both Llama generations no windowed placement, localizer-chosen or optimal, was ever the right decision at this budget: spreading was.

A.13 Load-Bearing McNemar Contrasts (Code)

The load-bearing subset of the frozen code-domain exact-McNemar family (§4.4); Figure 2 plots every frozen contrast in both per-item-tested domains.

Timeline and selection direction. Every contrast—including those involving the realized optimum—was enumerated and frozen before any per-item grading was inspected. The optimum is the only *selected* arm in the families (localizer windows and random draws are pre-designated), and selection biases its contrasts upward, in its own favor: de-inflating the optimum weakens placement certification but cannot rescue a localizer, since no localizer arm is selected on the graded items. The training-seed re-run measures the inflation directly (best code draw $0.520 \rightarrow 0.493$; §4.5).

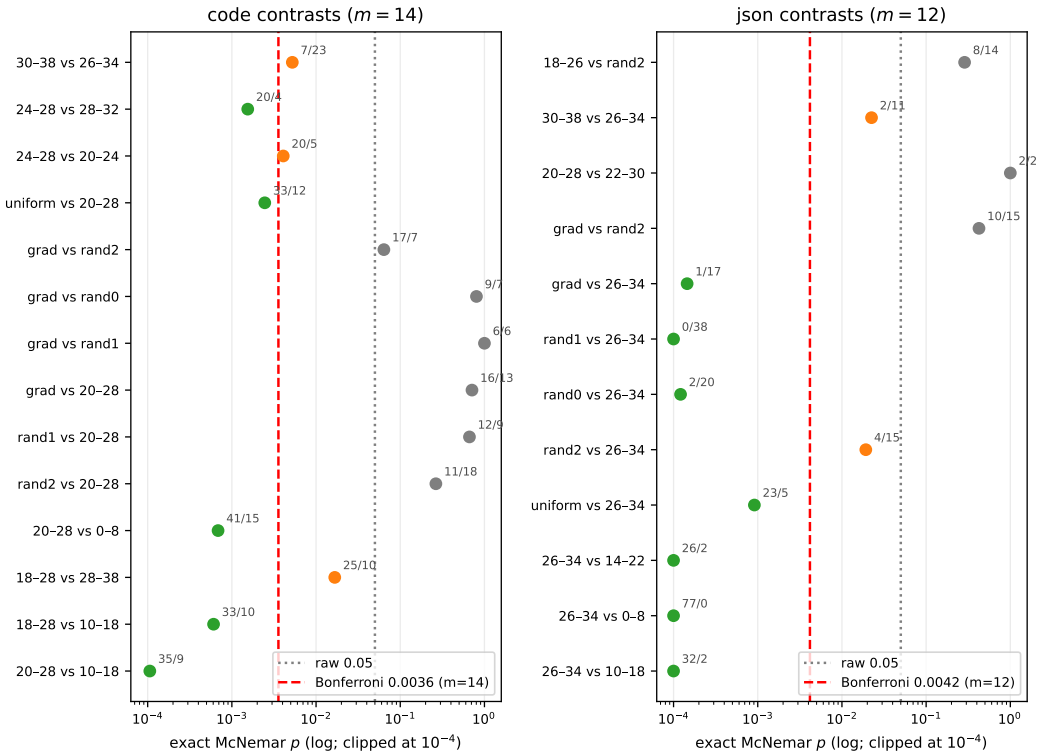

Figure 2: **Paired-significance forest: every frozen McNemar contrast, both per-item-tested domains.** Exact McNemar p (log scale, clipped at 10^{-4}); 14 code contrasts ($n = 300$, Bonferroni $\alpha/14 = 0.0036$) and 12 JSON contrasts ($n = 250$, $\alpha/12 = 0.0042$). Green = survives per-domain Bonferroni; orange = raw-significant only; gray = not significant; point labels give discordant-pair counts n_{10}/n_{01}. Gray *regret* contrasts are designated *nulls*: non-significance is the wanted outcome. Note the domain asymmetry: on code, the best window fails to separate from any random draw; on JSON, the c30 optimum beats two of three draws under Bonferroni—placement is statistically real, yet the localizer predictions still miss it. The code bridge contrast (c34 vs. c30) trends worse for c34 (raw $p = 0.0052$) but does not survive Bonferroni—unresolved within noise.

Table 5: Paired exact-McNemar contrasts on code per-item correctness ($n = 300$). n_{10}/n_{01} are discordant-pair counts. ✓ = survives Bonferroni ($p < 0.0036$); the regret contrasts are designated *nulls*—non-significance is the wanted outcome, not a shortfall.

Contrast	acc.	n_{10}/n_{01}	p	Reading
downstream(c24) vs. detection(c14)	.510/.423	35/9	0.0001	placement ✓
boundary vs. detection	.500/.423	33/10	0.0006	placement ✓
downstream vs. early	.510/.423	41/15	0.0007	placement ✓
uniform (full) vs. downstream	.580/.510	33/12	0.0025	efficiency ✓
top-gradient vs. downstream	.520/.510	16/13	0.71	*regret null*
random9 vs. downstream	.520/.510	12/9	0.66	*regret null*
random9′ vs. downstream	.487/.510	11/18	0.26	*regret null*

A.14 Minimum Detectable Effects and Effect Bounds

Non-significance is never equivalence (§4.4); this section quantifies what the ties do and do not bound. For each load-bearing tie we report the observed paired difference $\hat{\Delta} = (n_{10} - n_{01})/n$, its exact Clopper–Pearson 95% interval (conditional on the discordant count $n_d = n_{10} + n_{01}$), the one-sided 95% upper bound (UB) on the first arm's advantage, and the approximate minimum detectable effect at 80% power under the family's Bonferroni threshold, $\mathrm{MDE}_{80} \approx (z_{\alpha_{\mathrm{fam}}/2} + z_{0.80})\sqrt{n_d}/n$

(normal approximation to the conditional McNemar test). Two readings follow from Table 6. First, the headline ties are bounded, not merely unresolved: no gradient-localizer advantage over a random matched-budget draw exceeds $+5.7$pp (code, worst case across the three draws) or $+1.7$pp (JSON, vs. the best draw—the only draw contrast in that family) at one-sided 95% confidence; the Llama-3 windowed maximum's advantage over the best draw is below $+3.6$pp; AdaLoRA's upper bound on Llama-3 code is negative. On Qwen-JSON the discordant counts themselves ($n_d \leq 4$ of 250) mechanically cap any depth-analog-vs-draw placement effect at 1.6pp; on Qwen-code the corresponding MDE_{80} spans 5.5–7.6pp (m=5 families). Second, the families had power where it mattered: MDE_{80} for the tabulated ties spans 4.3–7.5pp, and the placement effects the grid *did* certify are of that order or larger—the tests could have resolved a practically meaningful localizer advantage had one existed.

Table 6: Effect bounds for the load-bearing ties. $\hat{\Delta}$ = paired accuracy difference (arm 1 − arm 2, pp); CI = exact 95% Clopper–Pearson interval; UB = one-sided 95% upper bound on arm 1's advantage; MDE_{80} = minimum detectable effect (pp) at 80% power under the family Bonferroni threshold. Draw indices as in Figure 2.

Contrast	n_{10}/n_{01}	$\hat{\Delta}$	95% CI	UB	MDE_{80}
Gemma code family (m=14, n=300)					
top-grad vs. downstream (c24)	16/13	+1.0	[−2.8, +4.6]	+4.1	6.7
top-grad vs. draw 0	9/7	+0.7	[−2.1, +3.2]	+2.9	5.0
top-grad vs. draw 1	6/6	0.0	[−2.3, +2.3]	+2.0	4.3
top-grad vs. draw 2	17/7	+3.3	[−0.2, +6.0]	+5.7	6.1
downstream (c24) vs. draw 1	9/12	−1.0	[−3.9, +2.2]	+1.8	5.7
downstream (c24) vs. draw 2	18/11	+2.3	[−1.5, +5.7]	+5.2	6.7
Gemma JSON family (m=12, n=250)					
optimum (c30) vs. best draw	15/4	+4.4	[+0.7, +6.7]	+6.5	6.5
top-grad vs. best draw	10/15	−2.0	[−5.8, +2.3]	+1.7	7.4
Llama-3 code family (m=10, n=300)					
windowed max (c14) vs. best draw	19/19	0.0	[−4.2, +4.2]	+3.6	7.5
AdaLoRA (c11) vs. best draw	11/22	−3.7	[−7.0, +0.4]	−0.2	7.0

A.15 Robustness Program Details

A.15.1 Training-seed variance

For each of twenty-eight load-bearing arms (four per model×domain cell on Gemma code, Gemma JSON, Qwen code, and Llama-3 code; all seven arms of the Qwen regex family; the five load-bearing arms of the Llama-3.1 code cell), the arm is retrained from scratch under two additional training seeds with data, steps, and hyperparameters unchanged, giving three runs per arm including the banked primary. Across all twenty-eight arms the per-arm seed standard deviation spans 0.002–0.021 (minimum: the Qwen-regex depth-fraction anchor; maximum: the Llama-3 windowed-maximum arm), against per-item binomial standard errors of 0.016–0.029. Table 7 reports each frozen contrast re-tested under every seed.

The landscape verdicts are therefore properties of the placement, not the optimization seed; and the numbers that fail to replicate are exactly the ones selection theory predicts should not—the post-hoc best draw's edge on Gemma code and the empirical maximum's edge over the strongest draw on Qwen regex. The Llama-3.1 anti-localization contrast runs *against* the selection gradient: the best window is itself a post-hoc maximum, so selection inflation favors it, yet its deficit to the best draw survives Bonferroni at every seed. On Llama-3 code, whole-network adaptation pays the grid's worst T_1 retention in every seed (0.780, 0.795, 0.755).

A.15.2 Ten random draws

The three frozen random matched-budget draws per Gemma domain are extended by seven additional draws (additive; the original three are unchanged), giving ten per domain. Code: draw accuracies span 0.457–0.520; the plateau optimum (0.510) lies inside the spread, and the mean draw regret is

Table 7: Per-seed replication of the frozen contrasts. "Sig." counts seeds (of 3) in which the paired McNemar test survives the per-domain Bonferroni threshold; "Dir." counts seeds in which the contrast points the same way as the primary (arm pairs named where role alone is ambiguous). Regret and Fisher rows are designated nulls: non-significance is the wanted outcome. The Gemma-code regret row also records the winner's-curse observation: the best random draw's primary-seed accuracy (0.520) falls to 0.493 under both re-seeds while the optimum stays within seed noise. Llama-3-code rows are graded against that cell's family threshold (0.005, m=10); its max-vs-random row is descriptive (the winner's-curse arm), where non-significance echoes the banked aggregate tie. Qwen-regex rows are graded at that cell's family threshold (0.0056, m=9); its max-vs-random rows are likewise the winner's-curse arm, reported per draw. Llama-3.1-code rows share the m=10 threshold; both arms of its anti-localization row are post-hoc maxima (re-seeding is a single-draw compare, not re-selection), and its band row was graded raw in the grid (post-hoc, outside the corrected family), so raw counts are reported there.

Contrast (role)	Cell	Sig.	Dir.	Note
placement	Gemma code	3/3	3/3	optimum > detection, every seed
bridge	Gemma code	3/3	3/3	sweep9_30to38 vs. downstream_20to28
placement	Gemma JSON	3/3	3/3	$p \approx 0$ every seed
placement	Qwen code	3/3	3/3	
placement	Llama-3 code	1/3	3/3	anchor loses to re-seeded draw every seed (p=0.0029/0.011/0.108)
uniform vs. best window	Llama-3 code	3/3	3/3	all $p \leq 0.0008$; worst retention every seed
anti-localization	Llama-3.1 code	3/3	3/3	best draw > best window every seed (p=0.0018/0.0002/0.0012)
uniform vs. best window	Llama-3.1 code	3/3	3/3	all $p < 10^{-4}$
band anchor vs. max	Llama-3.1 code	0/3	3/3	raw-sig. 3/3 (p=0.0095/0.0111/0.0295); graded raw in the grid
placement	Qwen regex	3/3	3/3	max vs. earliest (c30 vs. c14), $p \leq 0.0001$ every seed
anchor vs. earliest	Qwen regex	1/3	3/3	c24 vs. c14; echoes banked marginal (p=0.035 primary)
uniform vs. best window	Qwen regex	3/3	3/3	all $p < 10^{-4}$
regret (null)	Gemma code	0/3	flips	best draw 0.520 → 0.493, 0.493
regret (null)	Gemma JSON	1/3	3/3	optimum ahead every seed; sig. at seed 1 ($p = 0.0023$)
regret (null)	Qwen code	0/3	—	incl. one exact discordant tie (17/17)
max vs. random (descr.)	Llama-3 code	0/3	3/3	max at-or-above re-seeded draw; nominal p=0.049 at seed 1
max vs. random (descr.)	Qwen regex	3/3	3/3	vs. draw 1
max vs. random (descr.)	Qwen regex	3/3	3/3	vs. draw 2 (weakest)
max vs. random (descr.)	Qwen regex	1/3	3/3	vs. draw 3 (strongest): p=0.0020/0.146/0.0225
regret (null)	Qwen regex	0/3	3/3	anchor vs. draw 1; 0/3 raw
regret (null)	Qwen regex	3/3	3/3	anchor vs. draw 2 (weakest); anchor ahead every seed
regret (null)	Qwen regex	0/3	3/3	anchor vs. draw 3; anchor below, tie-adjacent every seed
Fisher null	Gemma JSON	0/3	—	downstream_22to30 vs. sweep9_26to34

+1.7pp with draws on both sides of the optimum. JSON: all ten draws score below the c30 optimum (0.840); the best draw (0.796) is the same window as the best of the original three and remains 4.4pp short. These distributions are reported descriptively; no draw-vs-optimum comparison in this expansion is graded by paired test (the per-seed McNemar replications of Table 7 carry significance).

A.15.3 CAPACITY TIER ($r = 32$)

Design: T_1 and all eleven code-domain arms are retrained at rank 32 ($\alpha = 64$; α/r fixed at 2.0) in both architectures. Four within-tier McNemar contrasts per model were specified before unblinding (three regret nulls, one placement), Bonferroni $\alpha/4 = 0.0125$. Ranks are distinct budget classes: no cross-rank contrast is ever graded by paired test. Arm-level training health at $r = 32$ (both carried-over optima, same 66 optimizer steps as their $r = 16$ counterparts): the Gemma arm trains 7,667,712 parameters (exactly 2× the $r = 16$ arm count of 3,833,856), final loss 0.5564, T_1 retention 0.950; the Qwen arm trains 5,046,272 (2× 2,523,136), final loss 0.865, T_1 retention 0.955. (The $r = 32$ T_1 finetunes themselves are full-budget retrains—e.g. Gemma T_1@$r32$: 108,036,096 trainable, loss 0.0171, 375 steps—and have no retention/step fields comparable to arms.)

Qwen $r = 32$ arm accuracies (Table 9), in rank order: whole-network 0.550; random draws 0.420, 0.407, 0.320; the declared depth-analog window 0.407 (exactly tying one draw); remaining principled windows 0.403, 0.400, 0.387, 0.383, 0.257, 0.210 (the two earliest windows at the bottom, mirroring the $r = 16$ early-window deficit). Gemma $r = 32$: whole-network 0.580 (unchanged from $r = 16$), windowed arms spanning 0.457–0.537 with the windowed maximum at the same window as the

Table 8: Within-tier $r = 32$ McNemar contrasts (pre-specified family, $m = 4$ per model, Bonferroni 0.0125). Regret rows are designated nulls. On Qwen the carried-over window is the depth-fraction analog of the r=16 optimum (declared arm), *not* the empirical $r = 32$ maximum—at $r = 32$ a random draw and whole-network both score above it.

Contrast	Model	p	Reading
optimum vs. draw 1	Gemma-2	0.2101 (11/5)	regret null holds
optimum vs. draw 2	Gemma-2	0.1671 (13/6)	regret null holds
optimum vs. draw 3	Gemma-2	0.0501 (22/10)	regret null holds (borderline raw; n.s.)
optimum vs. earliest	Gemma-2	0.0003 (33/9)	placement separates ✓
declared arm vs. draw 1	Qwen-2.5	1.0	exact tie (21/21 discordant)
declared arm vs. draw 2	Qwen-2.5	0.6516	draw scores above the declared arm
declared arm vs. draw 3	Qwen-2.5	2.4×10^{-5}	driven by one weak draw (0.320)
declared arm vs. earliest	Qwen-2.5	2.5×10^{-15}	placement separates ✓

Table 9: Full r=32 arm accuracies (code domain, both architectures, $n = 300$), from the banked r=32 eval runs. Whole-network tops every windowed arm in both models. The Gemma windowed maximum lands at the same window as the r=16 optimum (c24), with the runner-up (c34, 81% relative depth) one held-out item behind; the Qwen declared depth-analog arm (ctr17) exactly ties a random draw at 0.407.

Arm (Gemma / Qwen analog)	Gemma-2	Qwen-2.5
whole-network (UNIFORM_ALL)	0.580	0.550
c14 / ctr9	0.457	0.210
c18 / ctr12	0.473	0.257
c22 / ctr15	0.493	0.400
c24 / ctr16	**0.537**	0.403
c26 / ctr17 (Qwen declared arm)	0.527	0.407
c30 / ctr20	0.527	0.383
c34 / ctr23	0.533	0.387
random draw, seed 0	0.517	0.407
random draw, seed 1	0.513	0.420
random draw, seed 2	0.497	0.320

$r = 16$ optimum. Descriptively, the extra capacity is spent unevenly: Gemma windowed arms move by at most ∼3pp with whole-network unchanged (0.580), while Qwen whole-network gains 10pp (0.450 → 0.550) and two random draws gain 6–8pp against the carried-over window's +2.0pp—at $r = 32$ that window is no longer even the numerical maximum among Qwen window arms.

A.15.4 FULL-PRECISION CONTROL (BF16)

A different confound could manufacture the code plateau: all sweeps use 4-bit NF4 quantization, which compresses activation precision and could flatten placement differences that full precision would resolve. We therefore retrained the code-domain regret family at bf16 (the numerically stable full-precision choice for Gemma-2; the optimum, three random draws, and the detection window), holding all else fixed. The regret ties persist: the optimum does not significantly beat any random draw at bf16 (paired exact-McNemar, $p = 0.70$, 0.36, 0.64; discordant 12/15, 7/12, 19/23; all n.s.), so the flat plateau is a property of the task and budget, not a quantization artifact. As with the r=32 tier, non-significance is not equivalence; we claim only that no separation emerges. One contrast changes with precision, and we report it plainly: the detection-vs-downstream separation that survives Bonferroni under NF4 ($p = 0.0001$; Table 5) is not significant at bf16 ($p = 0.24$, discordant 28/19)—at full precision the code landscape is if anything flatter still, the early-window deficit itself falling within noise. This neither strengthens nor weakens the placement claim; it is reported as measured.

The same control extends to JSON, the peaked domain: the optimum, three random draws, and the detection window were retrained at bf16 ($n = 250$, $m = 4$). The no-certified-advantage

verdict is precision-robust—the optimum (0.868) beats no draw (0.856, 0.848, 0.856) even at raw thresholds ($p = 0.375$, 0.180, 0.508; at NF4 its edge over the best draw was already raw-significant only, $p = 0.019$)—and the peak-vs-detection separation that survives Bonferroni under NF4 is not significant at bf16 (0.868 vs. 0.840, $p = 0.092$). Full precision *compresses* this landscape rather than sharpening it: the no-T_2 floor rises from 0.264 to 0.312 and the detection window from 0.720 to 0.840. The licensed claim is therefore scoped: no certified advantage over random is precision-robust; landscape *shape* is not claimed to be precision-invariant. One protocol parameter necessarily co-varies with the tier—NF4 trains with a paged 8-bit optimizer, bf16 with full-precision optimizer states—so the control tests the precision stack as deployed, not weight dtype in isolation. Fixed-seed retraining on this stack reproduces per-item vectors only to within 1–4 items of 250 (base-model evaluation is bit-exact); repro checks on released artifacts are tolerance-based, never exact-reprint.

A.16 Depth Band: Construction and Grading

The depth band is the intersection of the top-2 regions of the three resolved placement landscapes, each expressed as a fraction of network depth, computed and banked *before* any subsequent full-grid landscape was unblinded. The intersection is non-empty: $[0.554,\ 0.595]$ of relative depth ($\approx$55–59%). Membership: the Gemma code optimum (c24, 0.571), a member of the Gemma JSON top-2 tie (c24), and the Gemma-analog of the Qwen code top-2 (center 16, 0.571) all fall inside; note the Qwen empirical maximum (center 17, 0.607) sits just outside, which is why the band is reported as a region hypothesis, not a point claim. Grading rule for any new full-grid landscape: locate the windowed maximum and report whether its depth fraction falls inside the frozen band, alongside the landscape's resolvability R. The $r = 32$ Gemma landscape grades CONSISTENT, NON-PROBATIVE under this rule: its windowed maximum (0.537) sits at depth 0.571 (inside), but the runner-up (0.533, depth 0.810, outside) trails by one held-out item, and the gap corresponds to $R \approx 0.1$—a resolution-limited flat top that cannot distinguish band membership from its absence. The Llama-3 8B code grid supplies the first resolvable out-of-sample grading ($R = 11.2$): the windowed maximum (center 14) sits at depth 0.438 and the runner-up (center 11) at 0.344—both outside the band—and the in-band window (center 18, depth 0.563) loses by paired McNemar to the out-of-band maximum ($p = 0.0015$) and to all three random draws ($p = 0.0029$, 0.0005, 0.0026; all Bonferroni-significant, $m = 10$); the loss to the re-seeded draw is direction-consistent in $3/3$ training seeds (Appendix A.15). The band grades DOES NOT GENERALIZE. No localizer in the battery predicted a window inside this band in any domain.

A.17 Gemma-2 Regex: Full Battery Detail

Full per-localizer detail for the flat Gemma-2 regex landscape summarized in §4.2. The windowed maximum (c24, 0.860) is statistically indistinguishable from all three random matched-budget draws (paired McNemar $p = 1.00$, 0.77, 0.049; none Bonferroni-significant); the only resolved landscape structure is an early-worse region (c24 vs. c14, $p = 0.0059$); and the whole-network ceiling (0.917) beats the best window (paired $p = 0.006$)—so no placement, localizer-chosen or optimal, recovers full-budget performance. Against this landscape the battery scores $0/7$: the two SAEs and the gradient-direction localizer name plateau windows within noise of the optimum (regret $+0.020$–0.030); Fisher's uniform reduction grades DIFFUSE while its band skeleton names c30—the reduction disagreement of §4.2, now in its starkest named-vs-diffuse form; the linear probe saturates at emergence layer 1 (probe-uninformative-by-saturation, as in code); AdaLoRA's allocation peaks at L41 and edge-snaps to c34 (miss, regret $+0.033$ vs. the random mean's $+0.012$); and AtP+GradDrop names c14—the single window significantly *below* the plateau. The random baseline's best draw ties the oracle exactly. On a flat landscape, hit/miss labels overstate—regret and the paired ties are the primary reading—with one exception: a localizer can still fail outright by steering into the resolved early-worse region, and the battery's most reliable localizer did.

A.18 Headroom Admissibility Audit

The complete audit table for §4.7: every attempted domain$\times$model combination, including those that never reached a placement sweep.

Table 10: Headroom admissibility for all sixteen attempted domain×model combinations (§3.4): floor p_f (base + T_1, no T_2 training), ceiling p_c (whole-network adaptation), both on the same n-item T_2 held-out set; $H = p_c - p_f$; z and R as defined in §3.4. H, z, and R are computed from unrounded proportions (Gemma QA: $0.3005 - 0.2732 = 2.7$pp; Qwen QA: $0.1913 - 0.1803 = 1.1$pp; Llama-3.1 QA: $0.2678 - 0.2295 = 3.8$pp). Verdicts are readings of z and R, not thresholds. The Qwen-2.5 rows are the cross-architecture tier, which replicates the code, JSON, and regex domains, plus QA as a cross-architecture admissibility check; the Llama-3 8B row is the out-of-sample test of the depth band on its original checkpoint, and the Llama-3.1 8B rows are the full third-architecture tier (all four domains; §5, Appendix A.10).

Domain	Model	p_f	p_c	n	H(pp)	z	R	Verdict
Code	Gemma-2 9B	0.247	0.580	300	33.3	8.8	11.7	gradable
JSON Schema	Gemma-2 9B	0.264	0.912	250	64.8	19.6	36.2	gradable
Factual QA	Gemma-2 9B	0.273	0.301	183	2.7	0.6	0.8	H within noise; ungraded
Regex dialect	Gemma-2 9B	0.070	0.917	300	84.7	39.0	53.1	gradable; resolved flat
JSON Schema	Qwen-2.5 7B	0.780	0.876	250	9.6	2.9	4.6	gradable
Code	Qwen-2.5 7B	0.177	0.450	300	27.3	7.6	9.5	gradable
Factual QA	Qwen-2.5 7B	0.180	0.191	183	1.1	0.3	0.4	H within noise; ungraded
Regex dialect	Qwen-2.5 7B	0.347	0.920	300	57.3	18.1	36.6	gradable; late plateau
Code	Llama-3 8B	0.190	0.513	300	32.3	8.8	11.2	gradable; max ties random
Code	Llama-3.1 8B	0.223	0.513	300	29.0	7.7	10.0	gradable; best draw beats best window
JSON Schema	Llama-3.1 8B	0.400	0.892	250	49.2	13.4	25.1	gradable; early-peaked cliff
Regex dialect	Llama-3.1 8B	0.033	0.963	300	93.0	62.0	85.7	gradable; strict early peak, largest H
Factual QA	Llama-3.1 8B	0.230	0.268	183	3.8	0.8	1.2	H within noise; ungraded
Pinyin	Gemma-2 9B	0.604	0.652	250	4.8	1.1	1.6	unresolvable at this n
Induction	Gemma-2 9B	—	—	—	—	—	—	died: base saturated (≥ 0.95)
Trad. Chinese	Gemma-2 9B	—	—	—	—	—	—	died: no shift (base $\approx T_1 \approx T_2$)

A.19 Split-Half Protocol Details

Each reliability leg recomputes a localizer's predicted window on an evaluation slice fully disjoint from its primary, enforced by a guard requiring offset $\geq n_{\text{primary}}$; a previously recorded leg at offset 50 against an $n = 100$ primary (50 shared examples) is rejected by this guard and disclosed below as a superseded working-record datum. Gemma primaries use $n = 100$; legs run at offset 100 (Fisher, probe), offset 30 at attribution level with target features fixed from the primary selection (both SAE widths, AtP+GradDrop), and offset 79 for QA on the T_2-only pool (79/79 fully disjoint). Qwen primaries use $n = 50$, forced by the smallest shared pool (QA T_1 eval $= 80$) under the same offset rule; Qwen T_2-train pool sizes are 350 (code), 207 (JSON), and 158 (QA), all file-backed before any leg was declared feasible. Pool consumption is itself reported: the Gemma probe's primary consumed the entire eval pool in JSON (100 of 100) and QA (80 of 80), so no disjoint leg exists there by construction—a data-appetite property of the method, recorded as such rather than as a missing cell (probe code regenerates from 100 of 200).

A.19.1 Gemma results

Stability without validity: SAE-16k is split-half stable in all three domains (circuit-profile cosine 0.98/0.84/0.91; top-5 layer overlap $3/5$, $4/5$, $3/5$)—yet against the grid it is diffuse or wrong. AtP+GradDrop is the benchmark's most reliable localizer—the identical window (c14, the grid floor) in every domain and every half—and that window is wrong everywhere placement matters: it is *reproducibly wrong*. The regex domain extends the anchor to a fourth domain (c14 again, band 1–11), and sharpens it: there c14 is the one window significantly below the plateau ($p = 0.0059$), so the most stable localizer in the battery reproducibly names the only measurably bad window. Fisher is stably a top-2 hit in JSON (c30 in both halves) and stably a miss in code (c34 in both halves). The linear probe is stably degenerate in code: both halves saturate at emergence layer 1, which our frozen rule reports as probe-uninformative-by-saturation, not as a window. Instability without a profile change: SAE-131k's window flips one grid step under resampling in JSON (c18 vs. c22) *while its top-5 layers agree* $5/5$—and again in QA (c22 vs. c18, also $5/5$). The underlying profile barely moves; the mechanical band-reduction rule amplifies sub-grid jitter across a snap boundary. The same

Table 11: Split-half reliability of each localizer's predicted window on fully disjoint data halves. SAE cells test *profile* stability with target features fixed from the primary selection (not feature-selection stability). QA is documentation-only (insufficient headroom; §3.4). Probe cells in JSON/QA are structurally absent: the probe's primary consumed the entire eval pool, so no disjoint half exists—reliability cannot even be measured. Note the Fisher-code cell reads DIFFUSE under the uniform scoring reduction of Table 1 but stable c34 under the skeleton reduction used here: the same instrument-sensitivity finding (§4.2), on the same cell.

Localizer	Code	JSON	QA (doc-only)
SAE-16k	stable (cos 0.98, $3/5$)	stable (cos 0.84, $4/5$)	stable (cos 0.91, $3/5$)
SAE-131k	stable (c22=c22)	**unstable** (c18 vs. c22; $5/5$)	**unstable** (c22 vs. c18; $5/5$)
AtP+GradDrop	stable (c14=c14)	stable (c14=c14)	stable (c14=c14)
Fisher	stable (c34=c34; miss)	stable (c30=c30; hit)	**unstable** (c34 vs. c30)
Grad-direction	stable-diffuse	stable-diffuse	stable-diffuse
Linear probe	stable (c14; degenerate)	no disjoint half	no disjoint half

Table 12: Split-half reliability of the portable localizers on Qwen-2.5-7B. The three SAE-dependent rows are structurally absent on this architecture (no pretrained SAE at the required coverage; AtP+GradDrop's target features are SAE-derived): nine N/A cells that restate the portability finding on the reliability axis. Across the graded cells, stable cells include hits, misses, and degenerate windows, while unstable cells include a localizer whose underlying layer ranking never moved—the class composition behind the body's no-signal-in-either-direction verdict. QA is documentation-only (ungraded landscape); JSON is unresolvable ($R = 4.6$), so validity adjectives apply on code only.

Localizer	Code	JSON	QA (doc-only)
Fisher	stable (c9=c9; floor arm)	stable-diffuse	stable (c23=c23)
Grad-direction	stable-diffuse	stable-diffuse	stable-diffuse
Linear probe	stable (degenerate)	no disjoint half	no disjoint half
SAE-16k / 131k / AtP+GD	N/A (toolchain absent on Qwen-2.5)		

reduction sensitivity documented against the grid (§4.2) thus also determines the *reliability* verdict: whether a localizer looks stable partly reflects the reduction rule, not the localizer. Meanwhile the gradient-direction localizer stably abstains everywhere, even as its top-5 layers churn completely in JSON ($0/5$ overlap)—the abstention is reliable, the underlying ranking is not. A previously recorded Fisher code flip (c34→c14) came from an offset-50 comparison whose halves share 50 of 100 examples; it was never a valid split-half, and the comparator's disjointness guard now rejects that leg. We disclose it as a superseded working-record datum, not a result. Fisher's QA instability lands, in the split half, on exactly the c30 the frozen Fisher QA band predicted—reproducing a working-record band discrepancy as sampling noise, and showing instability need not mean "wrong."

A.19.2 CROSS-ARCHITECTURE (QWEN-2.5) RESULTS

We repeat the split-half protocol on Qwen-2.5-7B for every localizer that ports (Fisher, gradient-direction, linear probe); the remaining rows are structurally absent on this architecture—no pretrained SAE at the required layer coverage and quality, and AtP+GradDrop's target features are SAE-derived—so the table's N/A pattern is itself the toolchain-portability finding of §5, now on the reliability axis. Table 12 gives the result (same disjointness guard and pool-feasibility gates as Table 11; primaries are $n = 50$, forced by the smallest shared pool—QA T_1 eval $= 80$—under the same offset $\geq n$ rule that gives Gemma $n = 100$).

Every runnable Qwen cell is stable, and none is thereby valid. On the one resolved Qwen landscape (code), all three stable outputs are demonstrably wrong or abstaining: Fisher stably names the floor arm (c9, second-worst on the grid), gradient-direction stably abstains, and the probe is stably degenerate (emergence layer 1 in both halves)—the same saturation replicated from Gemma, as is the probe's pool-consumption in JSON and QA. On Qwen-JSON and Qwen-QA only stability can be measured, not validity. The cross-architecture contrast completes the dissociation: Gemma

Table 13: Split-half reliability of the portable localizers on Llama-3.1-8B (same disjointness guard and pool-feasibility gates as Table 11). The SAE rows are N/A for a different reason than on Qwen-2.5: Llama Scope exists and its 32× width passes the reconstruction gate, but the placement-profile construction has no Llama analog and the 8× width fails the gate—availability ports, the placement construct does not. QA is documentation-only (ungraded landscape); its gradient legs' T_1 side is n=30, pool-truncated.

Localizer	Code	JSON	Regex	QA (doc-only)
Fisher	stable (c11=c11)	stable (c11=c11)	stable (c11=c11)	stable (c11=c11)
Grad-direction	stable-diffuse	stable-diffuse	stable-diffuse	stable-diffuse
Linear probe	stable (degenerate)	no disjoint half	no disjoint half	no disjoint half
SAE-16k / 131k / AtP+GD	N/A (placement profile not ported; 8× fails gate)			

exhibits genuine instability (SAE-131k's snap flips, Fisher's QA flip), while Qwen's portable battery is uniformly reliable and, wherever it can be checked, reliably wrong or abstaining. Stability is neither necessary nor sufficient for validity on either architecture.

A.19.3 Cross-architecture (Llama-3.1) results

The protocol repeats on Llama-3.1-8B across all four domains (Table 13). Unlike Qwen, the SAE rows are not absent for lack of an SAE: Llama Scope exists, and its 32× width passes the frozen reconstruction gate—but the placement-profile construction taps Gemma's post-feed-forward sandwich normalization, which has no Llama analog, and the 8× width fails the gate in every domain (Appendix A.10). Availability ports; the placement construct does not, and AtP+GradDrop remains SAE-tree-coupled. The three portable localizers give twelve cells: gradient legs run at offset 50 (the QA leg's T_1 side is truncated by its pool to n=30 of the requested 50, recorded as realized), and the probe's disjoint leg exists only on code—its n=100 primary consumes the binding T_1-eval pool elsewhere (100 of 100 on JSON and regex, 80 of 80 on QA), the same data-appetite property recorded on both other architectures.

Every runnable cell is stable, and the composition sharpens the dissociation to its extreme: Fisher names the identical window (c11) in both halves of all four domains—the same c11 its landscape-invariant readout emits everywhere on this architecture (§5)—while on the resolved code landscape that window certifies no advantage and the best window itself loses to a random draw. Maximal reliability and zero demonstrated validity, in the same cells. Gradient-direction stably abstains in all four domains; the probe is stably degenerate on code (saturation in both halves), replicating the Gemma and Qwen saturation. Reliability here is measured by this table's split-half comparator; the landscape-invariance of the same readouts is a cross-cell property measured by the localizer battery (§5)—the two findings compound, but they are distinct measurements from distinct instruments.

A.20 QA Dissociation Detail

Full offset-ladder detail for §4.7.

At matched adaptation budget, mid-layer windows absorb the T_2 distribution far better than late windows—a monotonic $\sim$10× decline in ΔNLL across the offset ladder (Figure 3, left panel; peak -0.611 at the center-22 window down to -0.054 at the latest, center-34 window, all at 2.13M parameters)—yet held-out accuracy is flat throughout (≈ 0.27–0.30, within one standard error). This is the sharpest single illustration that representational movement and decodable behavior can fully dissociate, and it is why localizer benchmarks must gate on the behavioral metric they claim to predict. The finding is architecturally robust: Qwen-2.5 7B on the same QA data yields $H = 1.1$pp ($z = 0.3$, $R = 0.4$), even weaker than Gemma's $H = 2.7$pp, and Llama-3.1 8B yields $H = 3.8$pp ($z = 0.8$, $R = 1.2$), so the not-decodable verdict holds across all three architectures (Table 10). On Llama-3.1 the dissociation itself replicates a third time: across the full ten-arm QA grid, every arm's gold-answer ΔNLL is negative (mean -0.40) with the same monotone depth gradient (center 11, $\sim$34% depth, -0.546, declining to -0.172 at center 23), while held-out accuracy stays flat (0.235–0.284, a random draw the nominal maximum). The ΔNLL magnitudes sit between Gemma's and Qwen's, and the reading is strictly descriptive on every architecture—NLL movement never

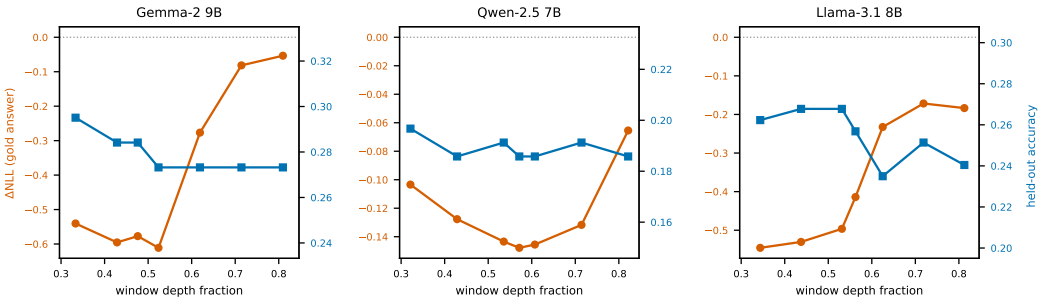

Figure 3: **The QA dissociation replicates on all three architectures.** Gold-answer ΔNLL (left axis) and held-out accuracy (right axis) over each architecture's QA windows, plotted against window depth fraction: Gemma-2 9B uses its offset ladder (5-layer windows, 2.13 M parameters); Qwen-2.5 7B and Llama-3.1 8B use their seven principled grid centers (random draws and the uniform arm carry no single center and are omitted, which is why the accuracy ranges here are narrower than the full-grid ranges quoted in the text; likewise the body's ΔNLL range quotes the full Gemma placement grid, whose boundary arms reach -0.883—the ladder plotted here troughs at -0.611). In every panel ΔNLL is negative at every window while held-out accuracy stays flat (Gemma 0.273–0.295, Qwen 0.186–0.197, Llama-3.1 0.235–0.268); the magnitudes differ by architecture (Gemma trough -0.611; Llama-3.1 -0.546 declining to -0.172; Qwen -0.065 to -0.148), but the shape—representational movement that never becomes decodable behavior—is common to all three, motivating the headroom admissibility criterion (§3.4) under which QA is ungraded. Strictly descriptive; NLL movement never grades a localizer.

grades a localizer. Figure 3 shows the three architectures side by side on a common depth-fraction axis.

A.21 Profile–Benefit Concordance

The frozen-rule rank-concordance table for §5. The strongest cell-level value is *negative* (Llama-3-code gradient-direction, $\tau=-0.81$, $p=0.011$, uncorrected across the eight cell-level tests). The instrument-level sign disagreement (two of the four resolved landscapes) means that even if some drift signal were placement-relevant, there is no consistent profile from which a practitioner could calibrate a correction.

A.22 A Cross-Cell-Calibrated Profile Readout (Eighth Localizer)

The seven localizers are per-cell decision rules: each reads one model–domain cell and names a window there. The eighth is a fixed cross-cell rule. For each held-out cell, a ridge regression ($\alpha=1.0$) is fit on the other eight graded cells to predict window benefit (held-out accuracy) from six features: the four window-mean localizer profiles (Fisher, gradient-direction, linear probe, AdaLoRA retained rank), each standardized within its cell, plus the depth fraction of the window center and its square. The held-out cell's predicted-benefit argmax (ties to the lower center) is the predicted window. The rule, the feature set, and every grading read below were fixed before any prediction was computed.

Comparators are graded on the same table: (i) the frozen depth band of Appendix A.16, operationalized as a predictor by snapping the band midpoint to the nearest grid center—the paper uses the band only as a membership test, so this placement operationalization is ours, likewise fixed before predictions; (ii) fixed depth fractions ($67\%/75\%$); (iii) AdaLoRA's own per-cell rule; and (iv) chance, which the tie-inclusive top-2 sets place at $\geq 2.57/9$ expected hits. Identification uses the battery's tie-inclusive top-2-distinct-values rule, byte-identical to the main tables.

Table 15 reports the record. The readout identifies a top-2 window in $8/9$ cells (sole miss: Llama-3.1 JSON, c20 vs. oracle c14); the band-as-predictor and fixed depth fractions each hit $4/9$, AdaLoRA's rule $3/9$, and exact binomial against chance gives $p\approx 3\times 10^{-4}$ (labeled approximate because tie-inclusive sets inflate the per-cell chance rate above $2/7$). The band's four hits are confined to the two model families it was frozen from (the three Gemma-2 cells and Qwen-code); it misses every

Table 14: **Profile–benefit concordance.** Kendall τ-b between each localizer's window-aggregated per-layer profile (mean over each grid window's layers, the same layer-to-window map the battery uses) and the seven-window benefit curve; exact n=7 permutation null, two-sided p in parentheses. Pooled: Stouffer Z across the four McNemar-resolved landscapes. Flat landscapes are reported but never pooled: the τ null conditions on the benefit ordering, and on flat landscapes the window spacings lie within evaluation noise, so concordance there is concordance with a noise draw. SAE/AtP profiles exist only on Gemma-JSON and are hook-capped: windows above the hook score zero and tie at the floor while benefit peaks above the cap (c30), so their negative signs are largely mechanical. The two SAE widths yield distinct layer profiles but induce the same ordering over the four sub-cap windows, hence their identical τ. The probe is excluded (saturation-degenerate in every domain except Gemma-QA, whose benefit curve is ungradable).

	Fisher	Grad-dir
Resolved landscapes (pooled)		
Gemma-JSON	+0.49 (0.18)	+0.59 (0.09)
Qwen-code	−0.29 (0.46)	+0.49 (0.18)
Qwen-regex	+0.29 (0.46)	+0.49 (0.18)
Llama-3 code	+0.62 (0.07)	−0.81 (0.011)
Pooled Stouffer Z (p)	+1.58 (0.11)	+0.91 (0.36)
Flat landscapes (reported, never pooled)		
Gemma-code	+0.68 (0.044)	+0.29 (0.46)
Gemma-regex	+0.88 (0.005)	+0.20 (0.66)
Single-cell, hook-capped (Gemma-JSON)		
SAE-16k	−0.69 (0.049)	
SAE-131k	−0.69 (0.049)	
AtP+GradDrop	−0.79 (0.017)	

Table 15: Cross-cell profile readout (eighth localizer): per-cell predictions and grading. Pred. = predicted window center; Oracle = best grid window; Top-2 = tie-inclusive top-2 distinct-value centers; an identification hit is a prediction inside that set. Band = frozen depth band as a predictor; 67/75% = fixed depth fractions; AdaLoRA = its per-cell rule. Cert. = certified advantage over matched random draws at the corrected bar. †: no traced correction family for this cell; the contrast fails at the raw bar regardless.

Cell	Pred.	Oracle	Top-2	Readout	Band	67/75%	AdaLoRA	Cert.
Gemma-2 code	c24	c24	{24, 30}	✓	✓	✓	×	×
Gemma-2 JSON	c24	c30	{24, 26, 30}	✓	✓	✓	✓	×
Gemma-2 regex	c26	c24	{24, 26, 30}	✓	✓	✓	×	×†
Qwen-2.5 code	c16	c17	{16, 17}	✓	✓	×	×	×
Qwen-2.5 regex	c20	c20	{17, 20}	✓	×	✓	×	✓
Llama-3 code	c11	c14	{11, 14}	✓	×	×	✓	×†
Llama-3.1 code	c11	c26	{11, 14, 26}	✓	×	×	✓	×
Llama-3.1 JSON	c20	c14	{11, 14}	×	×	×	×	×
Llama-3.1 regex	c14	c11	{11, 14}	✓	×	×	×	×
Total				8/9	4/9	4/9	3/9	1/9

Llama cell. At the corrected certification bar, one cell certifies: Qwen-regex, where the predicted c20 beats all three matched draws (α=0.0056; p=1.6×10^{-4}, 1.3×10^{-7}, 2.0×10^{-3})—the one landscape the battery had already marked as structurally open, and the plateau and selection caveats of Appendix A.10 apply unchanged. Gemma-JSON's predicted c24 beats two draws (p=7.4×10^{-3}, 7.9×10^{-9}) but not the best draw (0.796; p=0.42)—the same ceiling that blocked Fisher's identical window—and Llama-3.1 JSON's prediction significantly *loses* to all three draws.

Provenance and scope. A pooled leave-one-cell-out regression on the same banked grids (R^2=+0.31, against −0.12 for a depth-band baseline; within-cell permutation p=5×10^{-4}) showed that essentially all transferable signal is AdaLoRA's retained-rank profile; the static readouts (Fisher,

gradient-direction, probe) rank-track benefit *within* cells but do not transfer *across* them—the separation is specifically cross-cell linear transferability. A three-feature subset {AdaLoRA, depth, depth^2} identifies $9/9$; we report it as sensitivity only, since the subset was selected after seeing the pooled fit. Two disclosures bound the claim: the readout's *form* (features, α) was chosen from a pooled analysis of these same nine grids—leave-one-out holds out only the graded cell—and the rule consumes eight graded grids as calibration data. It is therefore a use case for the released benchmark, which is what makes cross-cell calibration possible at all, not a rescue of per-cell localization: on the anti-localization cells no window rule can certify, and the readout rightly fails certification there too.

A.23 Domain Details

Python: T_1 = Python 3.11 syntax (StarCoderData (Li et al., 2023)); T_2 = 3.12 features (PEP 695 type parameters, PEP 701 f-strings). **JSON Schema:** T_1 = draft-07; T_2 = draft-2020-12 (`prefixItems`, `$defs`, `$dynamicRef`, `if/then/else`). **Regex dialect:** T_1 = Python-dialect named groups (`(?P<name>...)`); T_2 = ECMAScript-dialect constructs (`(?<name>...)` named groups, `\k<name>` backreferences)—a dialect shift, not a parse failure. **Factual QA:** T_1 = pre-shift facts; T_2 = updated facts across leadership, sports, science, technology. Eval sizes n per domain as in Table 10; T_2 pools are split into train / held-out before experimentation (Appendix A.8).