# TARGET-DISTRIBUTION MATCHING FOR ACTIVATION STEERING: PAIRED CAUSAL RESIDUAL TRANSPORT

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Activation steering changes a frozen language model's behavior, but behavioral scores and hidden-state distances do not specify which output distribution it should produce. For instruction-defined control, the model supplies this target: its next-token distribution under the target instruction, given the same content and prefix. We introduce Paired Causal Residual Transport (PCRT), which predicts the same-content residual left by a fixed editor and adds it after final normalization, before the vocabulary head. The target instruction supplies training supervision and is absent at deployment. Within the learned residual subspace, the conditional mean minimizes squared output-head error, which bounds probability error before numerical rounding. On Gemma 2 2B, PCRT reduces 16-token sequence Kullback–Leibler (KL) divergence by 35.7% to 54.4% for Activation Transport (AcT) and Linear End-to-end Activation Steering (LinEAS), outperforming matched different-content residuals on all 64 test contents. AcT repair reduces 32-token sequence KL by 69.6% to 74.2% on Qwen 2.5, Qwen 3, and Llama 3.1. On five public instruction-following categories, direct correction lowers macro sequence KL by 22.2% and raises strict compliance by 44.1 percentage points. A fixed editor's remaining distributional error can therefore provide a learnable correction target, without retraining the editor.

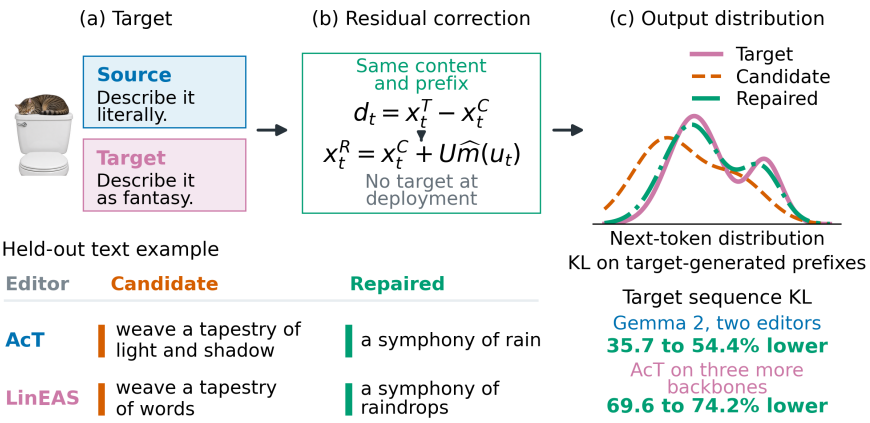


Figure 1: Paired Causal Residual Transport (PCRT) learns a fixed editor's same-content residual and predicts it without the target instruction at deployment. The target branch supplies supervision; the repaired branch reads the candidate state and observed prefix. The top row is schematic; the bottom row shows a separate held-out text example and reductions across all 64 test contents.

## 1 INTRODUCTION

Activation steering can change a frozen language model's behavior without changing its weights, but proxy-based evaluations often lack a shared definition of the correct output. Existing methods use contrastive directions, learned interventions, transport maps, or state-dependent fields (Turner



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.

et al., 2023; Li et al., 2023; Rimsky et al., 2024; Zou et al., 2023; Wu et al., 2024; Rodriguez et al., 2025a;b; Jin et al., 2026). Their hidden-state and behavioral measures show whether an intervention caused change. They do not specify the full conditional distribution that the intervention should reach. As a result, the apparent quality of an editor can depend on the proxy used to judge it.

For instruction-defined control, the frozen model itself provides a precise reference. Given the same content and response prefix, the target instruction induces a full next-token distribution. This observation leads to our central question: can an activation intervention reproduce that distribution when the target instruction is unavailable at deployment? Hidden proximity cannot answer this question because representation geometry depends on the chosen inner product, and behaviorally equivalent interventions can have different hidden descriptions (Venkatesh & Kurapath, 2026). Attribute scores discard most of the vocabulary distribution, while one decoded response is only one sample. A target distribution retains the information needed to compare different editors on the same output-level objective.

The required correction is not a single global direction. A fixed editor leaves different errors for different contents and response prefixes, so averaging those errors can erase the structure needed for repair. Moreover, equal hidden-state changes need not produce equal probability changes because the output head determines which directions are visible to the vocabulary. Autoregressive generation adds a final complication: each sampled token changes the next prefix. A useful correction must therefore depend only on the observed history and must be evaluated along generated histories, not at one fixed state.

Paired Causal Residual Transport, or PCRT, repairs this structured error from same-content trajectories (Figure 1). It represents residuals in the output-head geometry induced by the vocabulary matrix, estimates their conditional mean from the candidate state, and applies the correction after final normalization and before the vocabulary head. At each autoregressive step, the repair changes the current next-token distribution without rewriting the existing attention cache. The target state supplies supervision offline but is never read at deployment. This construction gives two testable predictions. Correctly paired residuals should outperform matched different-content residuals. Conditioning on the candidate should also outperform a paired residual mean when the remaining error is predictable from the candidate state.

Our experiments test this mechanism and its practical effect. On Gemma 2 2B and Common Objects in Context (COCO) captions, PCRT repairs both current-token kernels and target-generated sequence distributions for AcT and LinEAS (Gemma Team et al., 2024; Lin et al., 2014). Controls isolate the value of same-content pairing and candidate-dependent prediction. The post-editor result then reproduces across three additional backbones, while direct correction improves both distribution matching and strict Instruction-Following Evaluation (IFEval) compliance (Zhou et al., 2023). Together, these results show how an instruction-defined target can support the analysis, comparison, and repair of activation editors.

## 2 Related Work

Most activation-steering objectives describe a desired direction of change rather than a complete target distribution. Direction-based methods use prompts or labeled behaviors (Turner et al., 2023; Li et al., 2023; Rimsky et al., 2024), Representation Engineering studies population-level representations (Zou et al., 2023), and Representation Finetuning (ReFT) learns interventions in a frozen model, including low-rank transforms (Wu et al., 2024). AxBench compares such methods across concepts and behavioral measures and finds prompting to be a strong baseline under its protocol (Wu et al., 2025). Causal mediation and activation patching instead localize model components or test internal hypotheses (Vig et al., 2020; Zhang & Nanda, 2024). These approaches answer where or in which direction a model changes. PCRT asks whether the resulting output distribution matches a same-content prompted reference.

Richer steering methods expand the transformation class but still produce a candidate state that must be judged. Activation Transport maps activation populations (Rodriguez et al., 2025a), LinEAS learns multilayer distributional maps (Rodriguez et al., 2025b), Flow-based Activation Steering integrates concept-conditioned vector fields (Jin et al., 2026), and HyperSteer amortizes interventions across requests (Sun et al., 2025). Adaptive methods further vary the intervention with feedback,

context, language, or model-based control (Nguyen et al., 2026; Li et al., 2026; Hsu et al., 2026; Shi et al., 2026; Skifstad et al., 2026). PCRT requires an editor that exposes a candidate final state and paired target trajectories. We evaluate this composition with AcT and LinEAS. Its contribution is a shared target-distribution criterion and a correction for the residual that remains.

Paired prompt supervision is the closest source of training information. Context distillation transfers prompted behavior into model parameters (Snell et al., 2022; Ye et al., 2026), while Prompt Steering Replacement (PSR) learns activation-conditioned rank-one interventions from paired prompting trajectories (Heyman & Vandeputte, 2026). PSR learns to replace a prompt directly. PCRT instead begins with a fixed editor's candidate state, predicts the multidimensional residual left at its output interface, and evaluates the resulting full-vocabulary sequence distribution. Appendix F compares these supervision, intervention, deployment, and evaluation protocols.

## 3 PAIRED CAUSAL RESIDUAL TRANSPORT

### 3.1 PROBLEM SETTING AND OUTPUT INTERFACE

PCRT targets the conditional output distribution induced by a target instruction for a fixed source-to-target control task. It composes with an existing activation editor, whose output is the candidate to be corrected. Let $c$ denote content, $S(c)$ its source instruction, $T(c)$ the same content under a target instruction, and $y_{<t}$ the available response prefix. The target instruction supplies paired supervision during fitting and evaluation. It is absent when the fitted correction is deployed. A frozen model with editor $E$ produces the candidate final normalized state

$$x_t^C = F_{\theta,E}(S(c), y_{<t}), \qquad x_t^T = F_\theta(T(c), y_{<t}) \tag{1}$$

for the paired target state. Both evaluations share the content, prefix, and response scaffold. They differ in the source and target control instruction. We use *causal* in the non-anticipating autoregressive sense: every deployed feature depends only on the observed prefix.

The continuous model output interface determines which state differences affect the next-token distribution. For $x \in \mathbb{R}^d$,

$$a(x) = Wx, \qquad \ell(x) = h(a(x)), \qquad p(x) = \mathrm{softmax}(\ell(x)), \tag{2}$$

where $W \in \mathbb{R}^{V\times d}$ is the vocabulary matrix. Gemma 2 uses the coordinatewise transformation $h(a) = 30\tanh(a/30)$. The deterministic low-precision inference kernel is

$$\widetilde{p}(x) = \mathrm{softmax}(\widetilde{\ell}(x)). \tag{3}$$

Sequence results evaluate $\widetilde{p}$; the common-history diagnostic uses the scoring path in Appendix H. The repair acts after final normalization and before the output computation, preserving the existing attention cache. We write KL for Kullback–Leibler divergence and TV for total variation.

### 3.2 OUTPUT-HEAD GEOMETRY

The vocabulary matrix supplies an output-head quadratic form. We define

$$G = W^\top W, \qquad \|\delta\|_G^2 = \delta^\top G\delta = \|W\delta\|_2^2. \tag{4}$$

Directions in the null space of $W$ receive zero length because they cannot change the continuous output kernel. Thus $\|\cdot\|_G$ is a seminorm in general and a norm when $W$ has full column rank. We call the induced structure the output-head geometry.

**Proposition 1** (Probability error bound)**.** *Let $h$ be $L_h$-Lipschitz in Euclidean norm. For $\delta = x' - x$,*

$$\mathrm{KL}(p(x)\,\|\,p(x')) \le \frac{L_h^2}{4}\,\delta^\top G\delta, \tag{5}$$

$$\mathrm{TV}(p(x), p(x')) \le \frac{L_h}{2\sqrt{2}}\sqrt{\delta^\top G\delta}. \tag{6}$$

Both a linear head and the Gemma 2 softcap have $L_h \le 1$. Proposition 1 therefore connects $G$ to probability error for the continuous interface. It applies before rounding; empirical KL is computed directly under each study's declared scoring path. Appendix D isolates the numerical residual.

### 3.3 SAME-CONTENT RESIDUAL COORDINATES

The training pairs define the output-visible residual subspace. For training content $n$ and position $t$, let

$$d_{n,t} = x^T_{n,t} - x^C_{n,t}, \qquad \mathcal{S} = \mathrm{span}\{d_{n,t}\}_{\mathrm{train}}. \tag{7}$$

We construct $U \in \mathbb{R}^{d\times r}$ with

$$U^\top G U = I_r, \qquad \mathrm{span}(WU) \subseteq W\mathcal{S}. \tag{8}$$

The retained span uses a training-only floating-point backward-error criterion. Without truncation, its image equals $W\mathcal{S}$. Calibration and test data do not select $r$.

The candidate coordinate and its paired target correction are

$$u_{n,t} = U^\top G x^C_{n,t}, \qquad \Delta_{n,t} = U^\top G(x^T_{n,t} - x^C_{n,t}). \tag{9}$$

Correct pairing matters because $\Delta_{n,t}$ describes the correction for a particular content and prefix. A residual from another content need not target the same conditional kernel.

The residual basis separates two sources of output-head error. Let $P_U = WU(WU)^\top$ be the orthogonal projector onto the vocabulary-space image of the learned basis. Since $(WU)^\top WU = I_r$, every coordinate prediction $q \in \mathbb{R}^r$ satisfies

$$\|W(d_t - Uq)\|_2^2 = \|(I - P_U)Wd_t\|_2^2 + \|\Delta_t - q\|_2^2 . \tag{10}$$

The first term is residual error outside the retained span. The second is the prediction error inside that span. This identity turns conditional residual prediction into the exact controllable part of the output-head objective. Appendix D gives the short proof.

### 3.4 CONDITIONAL RESIDUAL PREDICTION

The population correction under squared head error is a conditional mean. Fix the fitted basis and assume $\mathbb{E}\|Wd\|_2^2 < \infty$. The common-history law fixes $t$ and samples content. The target-path law also samples target-generated histories and a uniform position in $\{1, \dots, L\}$, with $L = 16$ in the primary study and $L = 32$ externally. The feature $\phi$ uses candidate states available at that position. Define

$$m^*(\phi) = \mathbb{E}[\Delta \mid \phi]. \tag{11}$$

For every square-integrable measurable predictor $f$, the projection identity gives

$$\mathbb{E}\|\Delta - f(\phi)\|_2^2 = \mathbb{E}\|\Delta - m^*(\phi)\|_2^2 + \mathbb{E}\|m^*(\phi) - f(\phi)\|_2^2. \tag{12}$$

Thus $Uf(\phi)$ with $f = m^*$ minimizes expected $G$-visible residual error among corrections in the fixed learned residual span that are measurable from the chosen causal feature. The common-history study applies this definition separately at each position because its explicit prefix feature changes dimension. The target-path study uses one map under the declared mixture of training positions.

This projection also quantifies the value of candidate dependence. Let $\bar\Delta = \mathbb{E}[\Delta]$ denote the global paired residual mean. Combining Equations 10 and 12 gives

$$\mathbb{E}\|W(d - U\bar\Delta)\|_2^2 - \mathbb{E}\|W(d - Um^*(\phi))\|_2^2 = \mathbb{E}\|m^*(\phi) - \bar\Delta\|_2^2 \geq 0. \tag{13}$$

The population conditional predictor therefore weakly improves on the paired residual mean in expected output-head error. The gain is strict whenever the required residual changes predictably with the causal feature. This result does not depend on a linear estimator. Linear shrinkage is our finite-sample approximation to $m^*$. Equation 13 also makes the paired residual mean a nested population baseline. The corresponding control tests whether the fitted conditional predictor uses candidate-state variation beyond the global mean.

For $p = \dim(\phi)$, we fit a training-only linear shrinkage predictor,

$$\widehat{m}(\phi) = \bar\Delta + \widehat\Sigma_{\Delta\phi}\widetilde\Sigma^{-1}_{\phi\phi}(\phi - \bar\phi), \tag{14}$$

$$\widetilde\Sigma_{\phi\phi} = (1-\alpha)\widehat\Sigma_{\phi\phi} + \alpha\frac{\mathrm{tr}(\widehat\Sigma_{\phi\phi})}{p} I + \eta I. \tag{15}$$

The coefficient $\alpha \in [0, 1]$ is an analytic training-sample shrinkage ratio; $\eta > 0$ is the numerical stabilization in Appendix E. The sequence experiment uses the position-shared current-state feature $\phi_t = u_t$. The common-history analysis uses the causal-prefix feature $\phi_t = (u_1^\top, \ldots, u_t^\top)^\top$. The mean terms provide the affine intercept. Both features are prefix-measurable instances of the same conditional projection.

The deployed correction is

$$x_t^R = x_t^C + U\widehat{m}(\phi_t). \tag{16}$$

For the fitted target-path estimate, Equation 16 is an affine output-interface intervention of rank at most $r$. Appendix F writes this form explicitly. Its defining target is the residual left by a fixed editor for matched content and prefix. Since $x_t^C$ is a deterministic function of $(c, y_{<t})$, $\sigma(x_t^C) \subseteq \sigma(c, y_{<t})$. The current-state feature is therefore prefix-measurable in the stated sense, while its ability to predict the required residual remains an empirical property.

At each autoregressive step, the base editor produces $x_t^C$, PCRT forms $x_t^R$, and the original output transformation samples from the repaired kernel. The sampled token enters the next prefix. The target branch is absent.

**Proposition 2** (Reachable target recovery)**.** *If $Wd_t \in \mathrm{span}(WU)$ for $d_t = x_t^T - x_t^C$ and $\widehat{m}(\phi_t) = U^\top G d_t$, then $W(x_t^T - x_t^R) = 0$ and $p(x_t^T) = p(x_t^R)$.*

Proposition 2 describes reachable states for the continuous interface. General states can retain approximation, estimation, and low-precision residual error, which we measure directly under the deployed output kernel.

### 3.5 TARGET-PATH SEQUENCE CRITERION

Autoregressive factorization gives a sequence-level target for the repaired kernels. For $A \in \{T, C, R\}$, let $\widetilde{P}_A^{1:L}(\cdot \mid c)$ be the sequence distribution generated by the deployed kernel $\widetilde{p}_A$. Then

$$\mathrm{KL}(\widetilde{P}_T^{1:L} \| \widetilde{P}_A^{1:L}) = \sum_{t=1}^{L} \mathbb{E}_{Y_{<t} \sim \widetilde{P}_T} \mathrm{KL}(\widetilde{p}_T(\cdot \mid Y_{<t}, c) \,\|\, \widetilde{p}_A(\cdot \mid Y_{<t}, c)) \,. \tag{17}$$

The primary improvement for content $c$ is

$$D_{\mathrm{seq}}(c) = \mathrm{KL}(\widetilde{P}_T^{1:L} \| \widetilde{P}_C^{1:L}) - \mathrm{KL}(\widetilde{P}_T^{1:L} \| \widetilde{P}_R^{1:L}). \tag{18}$$

We estimate the expectation by sampling target response paths and evaluating all three full-vocabulary kernels on the identical target prefix. Target states provide training pairs and evaluation paths only. The deployed controller uses the candidate state and fitted parameters.

## 4 EXPERIMENTS

### 4.1 EVALUATION DESIGN

The evaluation asks whether PCRT repairs immediate output probabilities, whether the repair persists along generated histories, which components cause the gain, and whether the effect transfers beyond the primary setting. The primary study uses the frozen Gemma 2 2B instruction model, COCO captions, and two candidate editors, AcT and LinEAS (Gemma Team et al., 2024; Lin et al., 2014; Rodriguez et al., 2025a;b). The source asks for a literal scene description. The target asks for a vivid fantasy description while preserving the depicted content. A second attribute study requests a warm and appreciative factual tone and evaluates AcT in Appendix K.

Immediate repair compares candidate, repaired, and target kernels on the same response histories. Sequence repair then samples prefixes from the target distribution and evaluates all three full-vocabulary kernels on those shared prefixes. The two studies use disjoint train, calibration, and test splits. All fitted quantities are determined without test data. The main sequence estimate averages two target paths for each of 64 independent test contents over 16 response positions. Appendix H specifies the features, ranks, splits, and numerical implementation.

The external study keeps the COCO transformation fixed and changes the backbone to Qwen 2.5 7B, Qwen 3 4B, or Llama 3.1 8B (Qwen et al., 2024; Yang et al., 2025; Grattafiori et al., 2024). It tests

AcT post-editor correction and, separately, correction of the unedited source over 32 positions. A public-task study evaluates direct correction on Qwen 2.5 across five IFEval instruction categories, with 73 independent held-out items (Zhou et al., 2023). Appendix H.6 gives the complete splits and a matched-data all-layer Prompt Steering Replacement baseline with a mean-squared-error objective (A-PSR-MSE).

The inferential unit is an independent content or instruction item. Sequence studies use a wild bootstrap (Mammen, 1993) with 10,000 centered studentized Rademacher draws over item-level improvements and simultaneous intervals at nominal family level $0.05$ for each editor, model, or comparator family. Strict IFEval compliance uses the official evaluator and a stratified paired-item bootstrap. This design aligns the statistical unit with the deployment input.

### 4.2 PROBABILITY REPAIR ON SHARED HISTORIES

PCRT repairs the probability kernel left by both editors on shared response histories. It removes $79.5\%$ to $83.7\%$ of output-head error and $66.2\%$ to $71.0\%$ of exact post-softcap KL (Table 1 and Figure 2). Output-head error improves for every test content, and exact KL improves for at least 63 of 64 contents under each editor. Content-level bootstrap intervals exclude zero for all four mean reductions.

Table 1: PCRT removes most output-head and probability error on shared histories. KL uses a low-precision vocabulary head and full-precision softcap (Appendix H). Lower is better.

| | Output-head error (millions) | | | Exact KL | | |
|---|---|---|---|---|---|---|
| Editor | Candidate | PCRT | Removed | Candidate | PCRT | Reduced |
| AcT | 0.902 | 0.147 | 83.7% | 0.762 | 0.221 | 71.0% |
| LinEAS | 0.794 | 0.163 | 79.5% | 0.681 | 0.230 | 66.2% |

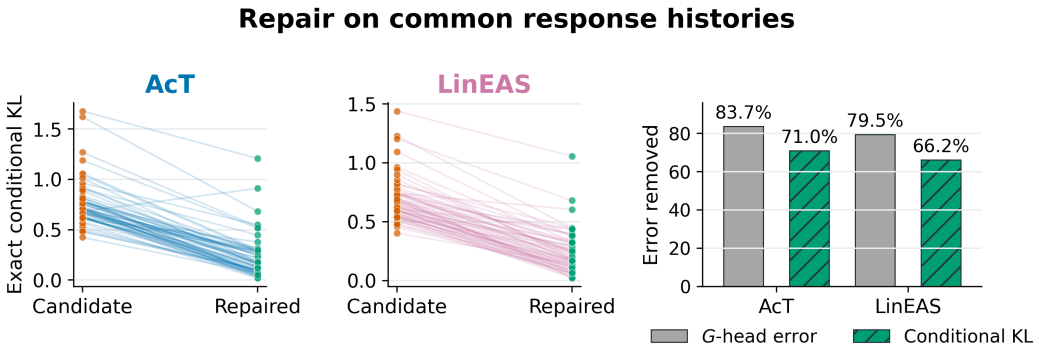


Figure 2: PCRT reduces output-head error and exact post-softcap KL for both editors on shared histories. Each line joins one test content before and after repair. The bar panel reports the fraction of mean error removed.

Figure 2 shows the content-level changes. The agreement between the $G$-error and exact probability KL is consistent with the probability-error bound. Remaining error can arise from target residuals outside the learned span and from conditional-mean estimation error.

### 4.3 REPAIR ALONG TARGET-GENERATED SEQUENCES

PCRT also brings the sequence distribution closer to its target. On target-generated prefixes, it lowers 16-token sequence KL by $54.4\%$ for AcT and $35.7\%$ for LinEAS (Table 2 and Figure 3). Both simultaneous intervals for the mean content-level improvement exclude zero. Because every conditional KL is evaluated on a prefix sampled from the target distribution, their sum estimates the forward sequence KL in Equation 17 rather than a fixed-prefix proxy.

Table 2: PCRT lowers target-path sequence KL over 16 response positions. Lower is better. Intervals are simultaneous 95% intervals for the mean improvement $D_{\text{seq}}$.

| Editor | Candidate | PCRT | Reduction | Improvement interval |
|---|---|---|---|---|
| AcT | 14.260 | 6.503 | 54.4% | [7.162, 8.352] |
| LinEAS | 10.080 | 6.478 | 35.7% | [3.205, 4.000] |

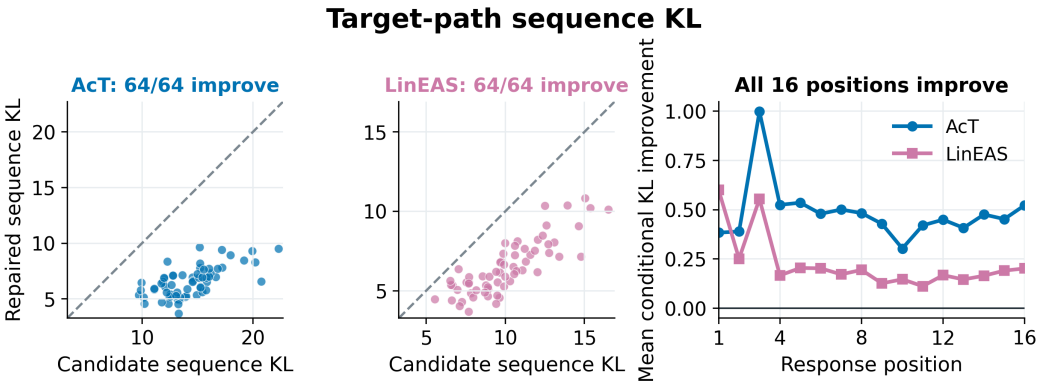


Figure 3: PCRT lowers target-path sequence KL for every test content under both editors. Points below the diagonal improve. The right panel shows that mean conditional KL improvement remains positive across all 16 response positions.

The primary sequence gain is consistent across contents and positions. With two target paths per content, every one of the 64 test contents improves under both editors, and mean conditional KL improvement is positive at all 16 response positions (Figure 3). Path-level counts and leave-one-content-out sensitivity in Appendix I give the same conclusion.

A second target attribute preserves the mean repair effect (Appendix K). AcT sequence KL falls by 26.8%, the simultaneous improvement interval is positive, 59 of 64 contents improve, and the mean gain remains positive at every response position. In this second attribute, the paired residual mean and conditional predictor are similar, so the evidence favors the value of same-content pairing over an added benefit from candidate conditioning.

Other probability views support the same conclusion. Total variation and pre-softcap logit error decrease for both editors on the same target paths, while sequence KL remains primary because the output transformation and vocabulary normalization weight errors differently from the quadratic bound. Appendix I defines and reports these secondary measurements.

### 4.4 MECHANISM CONTROLS

The controls identify same-content pairing and candidate dependence as effective components in the primary literal-to-fantasy task. Relative to a paired residual mean, the conditional predictor removes a further 19.6% of AcT's remaining sequence KL and 14.2% of LinEAS's. A harder control fits the same predictor class after replacing each training residual with a different-content donor residual selected by training-only matching on prompt length, response position, candidate entropy, and output-head residual norm. PCRT lowers sequence KL from 11.093 to 6.503 for AcT and from 9.947 to 6.478 for LinEAS. Both simultaneous gap intervals exclude zero, and PCRT is better on all 64 test contents for each editor. Figure 5 and Appendix I.6 report the complete comparison.

The output-head geometry explains the correction at the probability level. Replacing $G$ with matched Euclidean fitting changes mean head error by at most 0.55% when all other choices are fixed. Pairing and prediction therefore account for the observed gain, while $G$ defines output-visible risk and connects state error to probability error through Proposition 1.

Capacity alone does not explain the repair. Current-state and explicit-prefix features perform similarly in the tested linear comparison. Across ranks 128 to 1024, mean held-out output-head-error reduction remains positive, and lower ranks sometimes repair more despite covering less training residual energy. This pattern is consistent with lower prediction variance offsetting reduced span coverage. Appendix B reports the full analysis, which concerns continuous output-head error rather than sequence KL.

### 4.5 Repair across models and instruction-following tasks

The post-editor result reproduces across three additional backbones and two model families. PCRT lowers 32-token AcT sequence KL by $69.6\%$ to $74.2\%$ on Qwen 2.5, Qwen 3, and Llama 3.1 (Figure 4). Every one of the 64 shared contents improves on each backbone, all simultaneous improvement intervals exclude zero, and mean conditional KL improvement is positive at every response position. A separately fitted direct source correction also lowers sequence KL on all three models. Figure 4 and Appendix L report the model-level values.

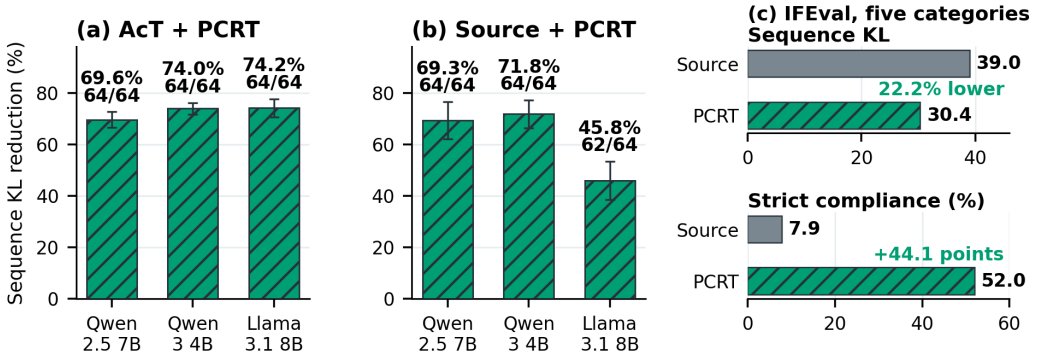


Figure 4: Target-distribution repair transfers across backbones and improves public instruction-following outcomes. The left and center panels report 32-token sequence KL reduction for AcT post-editor repair and direct source correction on the shared COCO transformation. Counts give contents with positive improvement out of 64. The right panel reports equal-category macro results over five held-out IFEval categories and 73 total items. Error bars show simultaneous $95\%$ intervals for the absolute mean KL gain, scaled by each observed candidate mean. The paired-bootstrap compliance interval appears in Appendix L.

Generated text supports the distributional result with a separate lexical check. Across AcT and LinEAS, repair raises source-caption unigram recall by $11.4$ to $14.6$ percentage points and paired-target unigram Jaccard by $9.7$ to $9.9$ points. Each candidate and repaired arm produces 64 distinct outputs. These checks measure lexical retention and output diversity. Semantic correctness and fluency remain separate properties. Appendix M defines the measures and gives representative outputs.

The direct source correction study connects distribution repair to strict instruction following. Across five held-out IFEval categories, macro sequence KL falls by $22.2\%$, and strict compliance rises from $7.9\%$ to $52.0\%$, a gain of $44.1$ percentage points. Matched PSR is stronger for direct prompt replacement under the same data protocol. PCRT addresses the complementary setting tested throughout this paper: low-cost repair of a candidate state produced by a fixed editor. Appendix L.4 reports every category and fitted seed.

PCRT adds little inference time. Adding it to AcT changes mean batch-one latency by $0.6\%$ to $1.7\%$ across the three external backbones, and no backbone parameter is updated. Appendix L.5 reports fitting time, stored parameters, latency, and batched throughput.

## 5 Discussion

In the Gemma 2 study, the controls consistently support same-content residual prediction. Correct pairing outperforms a different-content control after exact position matching and training-only rank

balancing on prompt length, candidate entropy, and output-head residual norm. A paired residual mean also repairs much of the target gap for both attributes. Candidate conditioning adds a clear gain for the primary attribute, where the edited state predicts part of the remaining residual, but little for the second. These nested controls yield a diagnostic. Paired-mean gain indicates a stable shift, and additional conditional gain indicates predictable content-specific error. Remaining error can reflect limited span coverage, prediction error, or unavailable information.

An editor's initial error does not determine how much of it can be corrected. AcT starts farther from the target than LinEAS, at $14.260$ versus $10.080$ mean sequence KL, yet repair reduces them to nearly identical values of $6.503$ and $6.478$. Candidate KL therefore mixes initial mismatch with recoverable structure. Reporting candidate and repaired KL separates these properties: candidate KL measures immediate accuracy, while the candidate-to-repaired gap measures structure recoverable by a shared predictor class. An editor with larger initial error can remain useful when that error is predictable and inexpensive to correct.

The distributional target changes both what can be learned and what can be evaluated. A task score records whether one decoded response obeys a constraint, while hidden-state distance measures an internal change. Neither shows how probability mass differs from the intended output. Full-vocabulary KL directly measures this difference, and its autoregressive decomposition identifies how mismatch changes along a response. The same target kernels supervise residual prediction during fitting and sequence KL on unseen contents. Behavioral scores and generated examples then answer a separate question: what consequences follow from the repair?

The output-head geometry and the sequence criterion play complementary roles. The geometry defines which state error is visible to current-token probabilities, separates subspace coverage from prediction, and yields a conditional-mean target. Matched Euclidean fitting performs similarly, so pairing and prediction account for the observed gain. Target-generated prefixes then extend the evaluation from one kernel to a forward sequence distribution shared by every editor. The theory defines the interface, while direct KL measurement establishes performance under the actual inference computation.

These findings position PCRT as a residual correction that composes with existing activation editors. Transport maps, learned editors, vector fields, and adaptive controllers produce candidate states through different online transformations (Rodriguez et al., 2025a;b; Jin et al., 2026; Li et al., 2026; Nguyen et al., 2026; Shi et al., 2026; Skifstad et al., 2026). PCRT takes such a candidate as its input, predicts the output-visible residual to a paired target, and repairs the state without retraining the editor. The closest paired-supervision alternative, Prompt Steering Replacement, instead learns direct prompt replacement (Heyman & Vandeputte, 2026).

The external results extend this objective beyond the primary architecture. Although the tested backbones differ in hidden dimension, vocabulary matrix, and output transformation, the same post-editor criterion improves every shared content. Direct correction also links lower sequence KL to higher strict compliance with little added online latency. This evidence supports a modular design: an editor produces the candidate state, and a conditional predictor repairs its remaining output-distribution error. Reporting the stages separately distinguishes initial editor error from residual predictability. Together with behavior-oriented evaluations such as AxBench (Wu et al., 2025), this design measures both behavioral change and distance from a specified output distribution.

Current evidence spans two editors, four backbones, two target attributes, and five public instruction categories over 16 or 32 tokens. Fully crossed editor-task studies and longer horizons remain open.

## 6 CONCLUSION

Instruction-defined control specifies the target for activation editing: the conditional output distribution induced by the target instruction. PCRT learns the same-content residual left by a fixed editor and predicts it from deployment-visible state. It repairs two editors on Gemma 2, transfers with AcT to three other backbones, and links lower sequence KL to stronger instruction compliance. Target distributions therefore provide a common probability-level objective for training, comparing, and improving activation interventions.

### AI USE STATEMENT

Generative AI tools supported mathematical formalization, experimental code checking, and manuscript polishing. The authors developed the theoretical framework and experimental design, interpreted the results, and verified the citations, equations, code, figures, insights, claims, and prose. The authors take responsibility for all content.

### ETHICS STATEMENT

Activation control can support transparent, task-specific model adaptation, but it can also make behavioral changes harder to audit. Responsible use should document the target instruction, paired data, intervention location, and evaluation scope. Our experiments use public, benign caption and instruction-following data and involve no human subjects or personal data.

### REPRODUCIBILITY STATEMENT

The complete experimental code and figure-generation scripts are provided in the Supplementary Material. Appendix H specifies the data split, prompts, output kernel, inference unit, statistical procedure, and controls. Appendices I, K, and L report the primary, cross-attribute, cross-backbone, public-task, and matched-comparator results. The single-cell program downloads public data and models, fits the editors and controllers, and runs all studies. It produces the result arrays used for tables, figures, and statistical summaries.

## REFERENCES

Gemma Team et al. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024. doi: 10.48550/arXiv.2408.00118. URL https://arxiv.org/abs/2408.00118.

Aaron Grattafiori et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. doi: 10.48550/arXiv.2407.21783. URL https://arxiv.org/abs/2407.21783.

Geert Heyman and Frederik Vandeputte. Steer like the LLM: Activation steering that mimics prompting. *arXiv preprint arXiv:2605.03907*, 2026. doi: 10.48550/arXiv.2605.03907. URL https://arxiv.org/abs/2605.03907. Accepted to ICML 2026.

Brandon Hsu, Daniel Beaglehole, Adityanarayanan Radhakrishnan, and Mikhail Belkin. Contextual linear activation steering of language models. *arXiv preprint arXiv:2604.24693*, 2026. doi: 10.48550/arXiv.2604.24693. URL https://arxiv.org/abs/2604.24693.

Zehao Jin, Ruixuan Deng, Junran Wang, Xinjie Shen, and Chao Zhang. Beyond steering vector: Flow-based activation steering for inference-time intervention. *arXiv preprint arXiv:2605.05892*, 2026. doi: 10.48550/arXiv.2605.05892. URL https://arxiv.org/abs/2605.05892.

Olivier Ledoit and Michael Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, 88(2):365–411, February 2004. doi: 10.1016/S0047-259X(03)00096-4. URL https://doi.org/10.1016/S0047-259X(03)00096-4.

Jiaqian Li, Yanshu Li, and Kuan-Hao Huang. Steering vector fields for context-aware inference-time control in large language models. *arXiv preprint arXiv:2602.01654*, 2026. doi: 10.48550/arXiv.2602.01654. URL https://arxiv.org/abs/2602.01654.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. In *Advances in Neural Information Processing Systems*, volume 36, pp. 41451–41530. Curran Associates, Inc., 2023. doi: 10.52202/075280-1797. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/81b8390039b7302c909cb769f8b6cd93-Paper-Conference.pdf.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *Computer Vision–ECCV 2014*, pp. 740–755. Springer, 2014. doi: 10.1007/978-3-319-10602-1_48. URL https://link.springer.com/chapter/10.1007/978-3-319-10602-1_48.

Enno Mammen. Bootstrap and wild bootstrap for high dimensional linear models. *The Annals of Statistics*, 21(1):255–285, 1993. doi: 10.1214/aos/1176349025. URL https://doi.org/10.1214/aos/1176349025.

Dung V. Nguyen, Hieu M. Vu, Nhi Y. Pham, Lei Zhang, and Tan M. Nguyen. Activation steering with a feedback controller. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2510.04309.

Qwen, An Yang, et al. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024. doi: 10.48550/arXiv.2412.15115. URL https://arxiv.org/abs/2412.15115.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. Steering llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15504–15522. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.828. URL https://aclanthology.org/2024.acl-long.828/.

Pau Rodriguez, Arno Blaas, Michal Klein, Luca Zappella, Nicholas Apostoloff, Marco Cuturi, and Xavier Suau. Controlling language and diffusion models by transporting activations. In *International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=l2zFn6TIQi.

Pau Rodriguez, Michal Klein, Eleonora Gualdoni, Valentino Maiorca, Arno Blaas, Luca Zappella, Marco Cuturi, and Xavier Suau. LinEAS: End-to-end learning of activation steering with a distributional loss. In *Advances in Neural Information Processing Systems*, volume 38, pp. 91527–91569, 2025b. doi: 10.52202/085713-2756. URL https://arxiv.org/abs/2503.10679.

Yingdong Shi, Ruiming Zhang, Changming Li, Zhiyu Yang, Kaixing Zhang, Jingyi Yu, and Kan Ren. UniSteer: Text-guided flow matching in activation space for versatile LLM steering. *arXiv preprint arXiv:2605.30076*, 2026. doi: 10.48550/arXiv.2605.30076. URL https://arxiv.org/abs/2605.30076.

Julian Skifstad, Xinyue Annie Yang, and Glen Chou. Local linearity of LLMs enables activation steering via model-based linear optimal control. *arXiv preprint arXiv:2604.19018*, 2026. doi: 10.48550/arXiv.2604.19018. URL https://arxiv.org/abs/2604.19018.

Charlie Snell, Dan Klein, and Ruiqi Zhong. Learning by distilling context. *arXiv preprint arXiv:2209.15189*, 2022. doi: 10.48550/arXiv.2209.15189. URL https://arxiv.org/abs/2209.15189.

Jiuding Sun, Sidharth Baskaran, Zhengxuan Wu, Michael Sklar, Christopher Potts, and Atticus Geiger. HyperSteer: Activation steering at scale with hypernetworks. *arXiv preprint arXiv:2506.03292*, 2025. doi: 10.48550/arXiv.2506.03292. URL https://arxiv.org/abs/2506.03292.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering language models with activation engineering. *arXiv preprint arXiv:2308.10248*, 2023. doi: 10.48550/arXiv.2308.10248. URL https://arxiv.org/abs/2308.10248.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Sohan Venkatesh and Ashish Mahendran Kurapath. On the non-identifiability of steering vectors in large language models. *arXiv preprint arXiv:2602.06801*, 2026. doi: 10.48550/arXiv.2602.06801. URL https://arxiv.org/abs/2602.06801.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart Shieber. Investigating gender bias in language models using causal mediation analysis. In *Advances in Neural Information Processing Systems*, volume 33, pp. 12388–12401. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/92650b2e92217715fe312e6fa7b90d82-Paper.pdf.

Zhengxuan Wu, Aryaman Arora, Zheng Wang, Atticus Geiger, Dan Jurafsky, Christopher D. Manning, and Christopher Potts. ReFT: Representation finetuning for language models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 63908–63962. Curran Associates, Inc., 2024. doi: 10.52202/079017-2041. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/75008a0fba53bf13b0bb3b7bff986e0e-Paper-Conference.pdf.

Zhengxuan Wu, Aryaman Arora, Atticus Geiger, Zheng Wang, Jing Huang, Dan Jurafsky, Christopher D. Manning, and Christopher Potts. AxBench: Steering LLMs? even simple baselines outperform sparse autoencoders. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 67035–67080. PMLR, 2025. URL https://proceedings.mlr.press/v267/wu25a.html.

An Yang et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. doi: 10.48550/arXiv.2505.09388. URL https://arxiv.org/abs/2505.09388.

Tianzhu Ye, Li Dong, Xun Wu, Shaohan Huang, and Furu Wei. On-policy context distillation for language models. *arXiv preprint arXiv:2602.12275*, 2026. doi: 10.48550/arXiv.2602.12275. URL https://arxiv.org/abs/2602.12275.

Fred Zhang and Neel Nanda. Towards best practices of activation patching in language models: Metrics and methods. In *The Twelfth International Conference on Learning Representations*. OpenReview.net, 2024. URL https://openreview.net/forum?id=Hf17y6u9BC.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023. doi: 10.48550/arXiv.2311.07911. URL https://arxiv.org/abs/2311.07911.

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, J. Zico Kolter, and Dan Hendrycks. Representation engineering: A top-down approach to AI transparency. *arXiv preprint arXiv:2310.01405*, 2023. doi: 10.48550/arXiv.2310.01405. URL https://arxiv.org/abs/2310.01405.

# A Appendix Contents and Scope

Table 3 organizes the appendix around mathematical results, implementation choices, and empirical evidence. The primary study tests two editors on Gemma 2, and a second study changes the target attribute. External studies keep the correction objective fixed while changing the backbone or moving to five public IFEval categories.

Table 3: Appendix contents.

| Section | Contents |
|---|---|
| B | Frozen-state rank and shrinkage sensitivity |
| C | Notation and the three probability kernels |
| D | Output-head bounds, causal projection, and exact recovery |
| E | Residual basis and analytic shrinkage estimator |
| F | Algebraic form and comparison with neighboring methods |
| G | Sequence KL identity, Monte Carlo estimator, and inference |
| H | Prompts, splits, numeric path, editors, external protocol, controls, and resource accounting |
| H.6 | Cross-backbone and public-task protocol |
| I | Full common-history and target-path results |
| I.6 | Matched different-content residual design and results |
| J | Position, path, content, calibration, and leave-one-out checks |
| K | Preregistered cross-attribute design, results, controls, and scope |
| L | Cross-backbone and public-task results |
| L.4 | Matched-data A-PSR-MSE comparison with three training seeds |
| L.5 | Fitting, storage, latency, and batched throughput |
| M | Generated-text diagnostics and representative outputs |
| N | Evidence reconstruction, numerical conformance, and audit scope |
| O | Limitations and remaining external scope |

The experiments evaluate prefix-measurable output-kernel correction across four frozen instruction models, two editors, two target attributes, a caption transformation, and five public instruction categories. The two-editor and cross-attribute studies use Gemma 2, the cross-backbone study uses AcT, and the public-task study uses direct correction on Qwen 2.5.

The theory and experiments address distinct parts of this scope. Proposition 2 characterizes a reachable continuous output kernel, while the sequence studies evaluate the deployed kernel and task-level outputs. The common-history study isolates correction under a shared numerical scoring path. The current state is an available deployment feature; our analysis does not claim that it is sufficient for the full prefix.

# B Rank and Shrinkage Sensitivity on Fixed States

The stored target-path states permit a direct sensitivity analysis of the finite-sample controller. Within the fitted full residual span, a training-only output-head principal-component rotation defines four nested ranks: 128, 256, 512, and 1024. These values equal $N/8$, $N/4$, $N/2$, and $N$ for $N = 1024$ training pairs. At every rank, we recompute the analytic training-sample shrinkage intensity $\alpha_*$ and evaluate fixed local factors $0.5\alpha_*$, $\alpha_*$, and $2\alpha_*$. Calibration and test outcomes select neither rank nor factor.

The analytic $\alpha_*$ yields positive mean held-out repair for both editors at all four ranks, and AcT improves all $64$ test contents throughout. Reduced ranks lower mean error for both editors even though they omit some training residual energy. This pattern is consistent across calibration and test, and it exposes the finite-sample tradeoff in Equation 10: greater span coverage can increase conditional-prediction variance.

The sensitivity analysis concerns the finite-sample mechanism rather than the primary sequence claim. The local factors do not establish predictive-risk optimality for the covariance shrinkage rule, and the predeclared sequence experiment retains its full-rank configuration. All quantities in Table 4 are continuous $G$-errors computed from stored states, not deployed KL, total variation, or generated behavior.

Table 4: Post hoc sensitivity of held-out continuous output-head error, distinct from sequence KL. Energy denotes the fraction of training residual energy represented by each nested rank. Cal. and test entries show the percentage of candidate error removed, with test values ordered as $0.5\alpha_*$, $\alpha_*$, and $2\alpha_*$.

| Editor | Rank | Energy | $\alpha_*$ | Cal. at $\alpha_*$ | Test at $0.5\alpha_*$ | Test at $\alpha_*$ / $2\alpha_*$ |
|---|---|---|---|---|---|---|
| AcT | 128 | 91.49% | 0.00852 | 63.71% | 64.30% | 64.88% / 65.57% |
| AcT | 256 | 95.65% | 0.00925 | 62.30% | 60.79% | 62.56% / 64.41% |
| AcT | 512 | 98.76% | 0.01015 | 57.01% | 52.52% | 57.72% / 61.78% |
| AcT | 1024 | 100.00% | 0.01127 | 47.43% | 41.85% | 51.69% / 58.44% |
| LinEAS | 128 | 87.67% | 0.00745 | 40.15% | 40.50% | 41.37% / 42.49% |
| LinEAS | 256 | 93.75% | 0.00808 | 36.93% | 34.87% | 37.35% / 40.11% |
| LinEAS | 512 | 98.22% | 0.00884 | 24.12% | 17.44% | 26.12% / 33.39% |
| LinEAS | 1024 | 100.00% | 0.00984 | 9.19% | −5.60% | 12.97% / 25.82% |

## C NOTATION

Table 5 collects the notation used in the method and experiment. Superscripts $T$, $C$, and $R$ always refer to the same target, candidate, and repaired probability kernels. They never denote different test contents.

Table 5: Notation used throughout the paper.

| Symbol | Meaning |
|---|---|
| $c$ | Content or instruction item, instantiated by one COCO caption or one IFEval prompt |
| $S(c), T(c)$ | Source and target instructions with identical content |
| $y_{<t}$ | Response tokens available before position $t$ |
| $x_t^C, x_t^T, x_t^R$ | Candidate, paired target, and repaired final normalized states |
| $W$ | Vocabulary output matrix with shape $V \times d$ |
| $h$ | Architecture-specific continuous output transformation |
| $p_A(\cdot \mid y_{<t}, c)$ | Continuous full-vocabulary conditional kernel for arm $A \in \{T, C, R\}$ |
| $\widetilde{\ell}_A, \widetilde{p}_A(\cdot \mid y_{<t}, c)$ | Deployed low-precision logits and conditional kernel for arm $A$ |
| $e(x) = \widetilde{\ell}(x) - h(Wx)$ | Deterministic deployment residual at state $x$ |
| $G = W^\top W$ | Positive-semidefinite output-head quadratic form on final normalized states |
| $d_t = x_t^T - x_t^C$ | Same-content target residual |
| $U$ | $G$-orthonormal basis for the retained output-visible training residual span |
| $u_t = U^\top G x_t^C$ | Candidate coordinate in the residual subspace |
| $\Delta_t = U^\top G d_t$ | Required target correction in residual coordinates |
| $\phi_t$ | Predeclared feature measurable from the candidate prefix |
| $m^*(\phi)$ | Study-specific conditional mean of $\Delta$ given $\phi$ |
| $\widehat{m}(\phi)$ | Training-only linear shrinkage estimate of $m^*(\phi)$ |
| $\widetilde{P}_A^{1:L}$ | Length-$L$ sequence distribution induced by $\widetilde{p}_A$ |
| $D_{\mathrm{seq}}$ | Candidate sequence KL minus repaired sequence KL |

The model follows the standard masked Transformer factorization (Vaswani et al., 2017). For every fixed content and instruction, the current normalized state is a deterministic function of the available prefix. This property gives prefix measurability. It does not give the reverse inclusion between the state and prefix sigma-fields.

# D Mathematical Details and Proofs

## D.1 The output-head probability bound

Let $C = I - \mathbf{1}\mathbf{1}^\top/V$ denote vocabulary centering. Softmax is invariant to a constant logit shift, so $\mathrm{softmax}(z) = \mathrm{softmax}(Cz)$. If $h$ is $L_h$-Lipschitz and $\delta = x' - x$, then

$$\|C(\ell(x') - \ell(x))\|_2 \le \|h(Wx') - h(Wx)\|_2 \tag{19}$$

$$\le L_h \|W\delta\|_2 = L_h\sqrt{\delta^\top G\delta}. \tag{20}$$

Define $f(z) = \log \sum_j \exp z_j$. Its gradient is softmax and its Hessian is

$$\nabla^2 f(z) = \mathrm{diag}(p) - pp^\top. \tag{21}$$

For any unit vector $v$, the quadratic form is the variance of the coordinate values $v_j$ under $p$. The range of a unit vector is at most $\sqrt{2}$, so this variance is at most $1/2$. Hence $f$ is $1/2$-smooth on the centered logit space. The softmax KL is the corresponding Bregman divergence,

$$\mathrm{KL}(\mathrm{softmax}(z) \| \mathrm{softmax}(z')) = f(z') - f(z) - \nabla f(z)^\top (z' - z). \tag{22}$$

Smoothness and Equation 20 yield

$$\mathrm{KL}(p(x) \| p(x')) \le \frac{1}{4}\|C(\ell(x') - \ell(x))\|_2^2 \le \frac{L_h^2}{4}\delta^\top G\delta. \tag{23}$$

Pinsker's inequality then gives

$$\mathrm{TV}(p(x), p(x')) \le \sqrt{\frac{1}{2}\,\mathrm{KL}(p(x) \| p(x'))} \le \frac{L_h}{2\sqrt{2}}\sqrt{\delta^\top G\delta}. \tag{24}$$

This proves Proposition 1.

## D.2 The deployed low-precision kernel

The continuous bound does not require low-precision rounding to be Lipschitz. Define the exact deterministic residual $e(x) = \widetilde{\ell}(x) - h(Wx)$ and let $C = I - \mathbf{1}\mathbf{1}^\top/V$. For $\delta = x' - x$, the triangle inequality gives

$$\left\| C\big(\widetilde{\ell}(x') - \widetilde{\ell}(x)\big) \right\|_2 \le L_h\sqrt{\delta^\top G\delta} + \left\| C\big(e(x') - e(x)\big) \right\|_2. \tag{25}$$

Applying the same log-sum-exp smoothness argument yields

$$\mathrm{KL}(\widetilde{p}(x) \,\|\, \widetilde{p}(x')) \le \frac{1}{4}\left( L_h\sqrt{\delta^\top G\delta} + \left\| C\big(e(x') - e(x)\big) \right\|_2 \right)^2. \tag{26}$$

Equation 26 separates the continuous head term from the measured deployment residual. It makes no smoothness assumption about rounding. The sequence experiments evaluate $\widetilde{p}$ directly, so their probability results do not depend on estimating the residual term.

## D.3 Why the nonlinear softcap remains inside the estimand

For Gemma 2, $h(a) = s\tanh(a/s)$ with $s = 30$. Its Jacobian is diagonal with entries $\mathrm{sech}^2(a_j/s)$, so its Euclidean Lipschitz constant is at most one. The softcap is coordinatewise nonlinear, however. In general,

$$h(a + c\mathbf{1}) - h(a) \notin \mathrm{span}\{\mathbf{1}\}. \tag{27}$$

A common shift can produce different coordinate changes because the softcap slope depends on the input. Removing the constant direction before the softcap can therefore remove a real probability change. Our quadratic risk uses $W^\top W$ as a bound before $h$, and every reported Gemma 2 KL includes the softcap before normalization, at the precision specified for its study. Qwen 2.5, Qwen 3, and Llama 3.1 use their native linear output transformation in the external study, so $h$ is the identity for those evaluations.

### D.4 CONDITIONAL PROJECTION

The exact head-error decomposition in Equation 10 follows from an orthogonal split in vocabulary space. Since $(WU)^\top WU = I$, the matrix $P_U = WU(WU)^\top$ is an orthogonal projector. For any $q$,

$$W(d - Uq) = (I - P_U)Wd + P_U Wd - WUq \tag{28}$$

$$= (I - P_U)Wd + WU\left(U^\top Gd - q\right). \tag{29}$$

The two terms are orthogonal. Taking squared norms and using the isometry of $WU$ proves Equation 10. Thus basis coverage and conditional prediction are separate error sources, and only the second depends on the controller within a fixed residual span.

Let $(\Delta, \phi)$ follow the study-specific law from Section 3. The assumption $\mathbb{E}\|Wd\|_2^2 < \infty$ also makes $\Delta = (WU)^\top Wd$ square integrable. Define $m^*(\phi) = \mathbb{E}[\Delta \mid \phi]$, and let $f(\phi)$ be square integrable. Add and subtract $m^*(\phi)$:

$$\mathbb{E}\|\Delta - f(\phi)\|_2^2 = \mathbb{E}\|\Delta - m^*(\phi)\|_2^2 + \mathbb{E}\|m^*(\phi) - f(\phi)\|_2^2 \tag{30}$$

$$+ 2\mathbb{E}\left[(\Delta - m^*(\phi))^\top (m^*(\phi) - f(\phi))\right]. \tag{31}$$

The cross term is zero after conditioning on $\phi$. Equation 12 follows. This result identifies the optimal correction $Uf(\phi)$ within the fixed learned residual span and chosen information set. The information set need not contain the full prefix.

Equation 13 is the same projection identity with the constant predictor $f(\phi) = \bar{\Delta}$. The output-span term from Equation 10 is identical for the two controllers and cancels. The remaining risk difference is

$$\mathbb{E}\|\Delta - \bar{\Delta}\|_2^2 - \mathbb{E}\|\Delta - m^*(\phi)\|_2^2 = \mathbb{E}\|m^*(\phi) - \bar{\Delta}\|_2^2. \tag{32}$$

Thus conditioning has zero population value exactly when the conditional mean is almost surely constant under the chosen feature.

### D.5 CAUSALITY OF THE DEPLOYED CORRECTION

Autoregressive masking makes $x_t^C$ a deterministic measurable function of $(S(c), y_{<t})$. The current-state coordinate $u_t = U^\top G x_t^C$ and every fitted affine operation on $u_t$ are therefore measurable with respect to the same available prefix. For the common-history feature, each additional coordinate $u_j$ has $j \le t$. Equation 16 reads no $y_j$ with $j \ge t$. Changing future tokens while holding the available prefix fixed cannot change $x_t^R$.

The intervention occurs after the final normalization. It changes the current output kernel, but it does not write corrected keys or values into the existing cache. At the next position, the sampled token enters the ordinary autoregressive prefix and the editor produces a new candidate state. This execution defines a valid sequence of repaired conditional kernels.

### D.6 EXACT RECOVERY FOR REACHABLE TARGETS

Let $d = x^T - x^C$. By Equation 8, $WU$ has orthonormal columns. If $Wd \in \mathrm{span}(WU)$, then its coordinate in this basis is

$$(WU)^\top Wd = U^\top W^\top Wd = U^\top Gd. \tag{33}$$

When $\widehat{m}(\phi) = U^\top Gd$,

$$Wx^R = Wx^C + WU\widehat{m}(\phi) = Wx^C + Wd = Wx^T. \tag{34}$$

Every deterministic continuous output transformation then produces identical logits and probabilities. This proves Proposition 2. Equality of $Wx^R$ and $Wx^T$ also gives deployed equality whenever the numerical output primitive receives the same pre-softcap tensor. We make no broader finite-precision claim from null-space equality alone. The condition can fail through residual components outside the retained span or through controller prediction error, and the deployed test KL measures the resulting probability difference directly.

### D.7 Algebraic invariance is narrower than semantic preservation

If $v^\top U = 0$, Equation 16 implies $v^\top x^R = v^\top x^C$. This is an algebraic coordinate invariant. It does not establish that every semantic property outside the requested target is unchanged. Semantic and factual preservation require task-specific evidence. We use same-content targets and report content-overlap diagnostics, while keeping the primary claim at the probability-distribution level.

## E Basis Construction and Linear Shrinkage Estimator

### E.1 Training-only output-visible basis

Let $D \in \mathbb{R}^{N\times d}$ stack the training residuals as rows. The implementation first computes an economy singular value decomposition of $D$ and retains the Euclidean row-space basis $Q \in \mathbb{R}^{d\times q}$. A singular value is retained only when it exceeds

$$\epsilon_{32} \max(N, d)\sigma_{\max}(D), \tag{35}$$

where $\epsilon_{32}$ is the unit scale of 32-bit floating-point arithmetic. This threshold is tied to numerical backward error rather than calibration performance.

The head-visible Gram matrix is $H = Q^\top G Q$. Write the retained eigendecomposition as $H = R\Lambda R^\top$. Eigenvalues use the analogous floating-point error threshold. The final basis is

$$U = QR\Lambda^{-1/2}. \tag{36}$$

It satisfies $U^\top G U = I$ up to the checked numerical residual. This two-stage construction drops null directions and directions below either numerical threshold. The retained output-visible span can therefore be smaller than $W\mathcal{S}$.

### E.2 Primal form of the estimator

Let $X \in \mathbb{R}^{N\times p}$ stack centered causal features and let $Y \in \mathbb{R}^{N\times r}$ stack centered residual coordinates. Define

$$\widehat{\Sigma}_{xx} = \frac{1}{N}X^\top X, \qquad \widehat{\Sigma}_{yx} = \frac{1}{N}Y^\top X, \qquad \mu = \frac{\operatorname{tr}(\widehat{\Sigma}_{xx})}{p}. \tag{37}$$

The fitted coefficient is

$$B = \widehat{\Sigma}_{yx}\left((1-\alpha)\widehat{\Sigma}_{xx} + \alpha\mu I + \eta I\right)^{-1}, \tag{38}$$

where $\eta$ is a machine-scale positive jitter. Prediction uses $\widehat{m}(\phi) = \bar{\Delta} + B(\phi - \bar{\phi})$.

### E.3 Analytic training-sample shrinkage

We use linear covariance shrinkage toward the scaled identity (Ledoit & Wolf, 2004). For centered rows $x_i$ and $\widehat{\Sigma}_{xx} = X^\top X/N$, its two sample quantities are

$$a = \left[\frac{1}{N^2}\sum_i \left\|x_i x_i^\top - \widehat{\Sigma}_{xx}\right\|_F^2\right]_+, \qquad b = \left\|\widehat{\Sigma}_{xx} - \mu I\right\|_F^2. \tag{39}$$

The implementation computes the same quantities in the sample-space dual to avoid an unstable inverse when the feature dimension is large. Let $K = XX^\top/N$ and $\mu = \operatorname{tr}(K)/p$. Since $\|K\|_F = \|\widehat{\Sigma}_{xx}\|_F$, it sets

$$a = \left[\frac{\sum_i \|x_i\|_2^4 - N\|K\|_F^2}{N^2}\right]_+, \tag{40}$$

$$b = \left[\|K\|_F^2 - p\mu^2\right]_+, \tag{41}$$

$$\alpha = \operatorname{clip}_{[0,1]}\left(\frac{a}{\max(b, 10^{-30})}\right). \tag{42}$$

The numerator measures training-row covariance dispersion, and the denominator measures departure from an isotropic target. Positions from one content are dependent, so we use this as a training-only regularization rule without invoking an independent-row covariance-risk guarantee. No calibration or test response enters $\alpha$. A small jitter

$$\eta = \max\{10^{-10}, \epsilon_{64} \max(\mu, 1)p\} \tag{43}$$

ensures a finite solve. The dual coefficient is

$$B = \frac{1}{N} Y^\top \left((\alpha\mu + \eta)I + (1-\alpha)K\right)^{-1} X, \tag{44}$$

which is algebraically equivalent to Equation 38.

### E.4 Chosen feature classes

The common-history mechanism analysis fits a separate map at each of its five positions and uses the explicit causal prefix feature

$$\phi_t = (u_1^\top, \ldots, u_t^\top)^\top. \tag{45}$$

The target-path sequence experiment pools all 16 positions from every training content and fits one position-shared map with $\phi_t = u_t$. Equal rows per content and position induce the uniform position mixture used in the population definition. The final state is already a function of the full available token prefix. A held-out ablation gives similar results for the two feature classes: the pooled current-state map removed $84.43\%$ and $79.13\%$ of head error for AcT and LinEAS, while the explicit-prefix mechanism result removed $83.68\%$ and $79.51\%$. This comparison supports the lower-dimensional deployed feature without establishing that the current state is sufficient for every history-aware prediction rule.

## F Algebraic Form and Relation to Neighboring Methods

The target-path controller is an affine activation intervention with a paired residual target. Write its fitted map as

$$\widehat{m}(u) = \bar{\Delta} + B(u - \bar{u}). \tag{46}$$

Substitution into Equation 16 gives

$$x_t^R = x_t^C + U(\bar{\Delta} - B\bar{u}) + UBU^\top G x_t^C. \tag{47}$$

The update lies in the $r$-dimensional residual range. Equation 47 makes the algebraic relationship to learned linear interventions explicit. The distinguishing supervision target is the remaining target residual of a fixed editor for the same content and prefix. The common-history analysis retains the same output range but uses the concatenated observed prefix coordinates as its feature.

Table 6 compares how neighboring methods obtain supervision, form candidate states, and define deployment targets (Wu et al., 2024; Hsu et al., 2026; Heyman & Vandeputte, 2026; Snell et al., 2022; Ye et al., 2026; Jin et al., 2026; Li et al., 2026; Shi et al., 2026; Skifstad et al., 2026). Prompt Steering Replacement is the closest supervision design because it also uses same-content prompted trajectories, but it learns direct prompt replacement. Other learned interventions and adaptive controllers construct candidate states through representation transforms, conditional flows, score gradients, inversion, or feedback. PCRT holds the editor fixed, predicts the same-content residual left on shared prefixes, and evaluates the resulting full-vocabulary sequence law. The table compares protocol roles; Appendix L.4 gives the matched empirical comparison.

## G Target-Path Sequence Estimator

### G.1 Chain-rule identity

For a length-$L$ sequence $y_{1:L}$,

$$\widetilde{P}_A^{1:L}(y_{1:L} \mid c) = \prod_{t=1}^{L} \widetilde{p}_A(y_t \mid y_{<t}, c). \tag{48}$$

Table 6: Supervision and deployment protocols. "Frozen" denotes an unchanged backbone.

| Method | Supervision unit | Intervention rule | Backbone | Deployment and primary target |
|---|---|---|---|---|
| ReFT | Task inputs and labels | Learned representation transforms, including low-rank interventions | Frozen | No teacher prompt, task performance |
| Contextual Linear Activation Steering (CLAS) | Contextual steering examples | Context-conditioned scale on a steering direction | Frozen | No target prompt, likelihood and behavior |
| Flow-based Activation Steering (FLAS) | Positive prompt, concept, and output triples | Concept-conditioned velocity field with multi-step Euler integration | Frozen | Condition retained, constructs an online candidate state for concept control |
| Steering Vector Fields (SVF) | Preference triplets flattened into binary continuations | Gradient of a learned concept scorer in a shared cross-layer space | Frozen | Updates candidate states at selected layers for context-dependent control |
| UniSteer | Activation-condition tuples indexed by layer and token position | Text-conditioned flow-matching vector field | Frozen | Source inversion followed by target-conditioned flow constructs an edited candidate state |
| Model-based linear optimal control | Positive and contrastive prompt sets, no policy training | Layer-wise feedback from local Jacobians and linear feature setpoints | Frozen | Updates realized states online to track semantic setpoints |
| Prompt Steering Replacement | Same-content source and target prompts on target trajectories | Activation-conditioned token-level prompt replacement | Frozen | Target prompt removed, imitate prompted behavior |
| Context distillation | Teacher outputs under richer context | Model parameters | Updated | Rich context removed, likelihood and task behavior |
| PCRT | Fixed-editor candidate and same-content target on shared prefixes | Affine output-interface residual correction of rank at most $r$ | Frozen | Target prompt removed, full-vocabulary target-sequence KL |

Expanding the log density ratio under $\widetilde{P}_T$ gives

$$\mathrm{KL}(\widetilde{P}_T^{1:L} \| \widetilde{P}_A^{1:L}) = \mathbb{E}_{Y_{1:L} \sim \widetilde{P}_T} \sum_{t=1}^{L} \log \frac{\widetilde{p}_T(Y_t \mid Y_{<t}, c)}{\widetilde{p}_A(Y_t \mid Y_{<t}, c)} \tag{49}$$

$$= \sum_{t=1}^{L} \mathbb{E}_{Y_{<t} \sim \widetilde{P}_T} \mathrm{KL}(\widetilde{p}_T(\cdot \mid Y_{<t}, c) \| \widetilde{p}_A(\cdot \mid Y_{<t}, c)) \,. \tag{50}$$

This is Equation 17. Sampling target paths performs Monte Carlo integration of an exact autoregressive identity. It does not replace the full-vocabulary kernel with a lexical score or model judge.

### G.2 Content-level estimator

For test content $c_i$, draw two independent paths $Y_i^{(r)} \sim \widetilde{P}_T^{1:L}(\cdot \mid c_i)$ for $r \in \{1, 2\}$. Define

$$Z_{i,A}^{(r)} = \sum_{t=1}^{L} \mathrm{KL}\Big(\widetilde{p}_T(\cdot \mid Y_{i,<t}^{(r)}, c_i) \| \widetilde{p}_A(\cdot \mid Y_{i,<t}^{(r)}, c_i)\Big) \tag{51}$$

and

$$\widehat{D}_i = \frac{1}{2} \sum_{r=1}^{2} \left( Z_{i,C}^{(r)} - Z_{i,R}^{(r)} \right), \qquad \widehat{D} = \frac{1}{n} \sum_{i=1}^{n} \widehat{D}_i. \tag{52}$$

The controller is fixed from training data. Conditional on those fitted parameters and the test contents, target-path sampling is independent across $r$ and

$$\mathbb{E}[\widehat{D} \mid \mathcal{F}_{\text{train}}, c_{1:n}] = \frac{1}{n}\sum_{i=1}^{n} D_{\text{seq}}(c_i). \tag{53}$$

The two-path average reduces Monte Carlo variance. The independent content remains the inference unit.

### G.3 Studentized wild-bootstrap interval

Let $d_i = \widehat{D}_i$, $\bar{d} = n^{-1}\sum_i d_i$, $e_i = d_i - \bar{d}$, and $\widehat{\text{se}} = s_d/\sqrt{n}$. For each bootstrap draw $b$, sample independent Rademacher signs $\xi_i^{(b)}$ and form

$$T_b^* = \left| \frac{n^{-1}\sum_i \xi_i^{(b)} e_i}{s(\xi^{(b)} \odot e)/\sqrt{n}} \right|. \tag{54}$$

We use 10,000 draws and the upper empirical quantile at $1 - 0.025$. The resulting interval is

$$[\bar{d} - q^*\widehat{\text{se}},\ \bar{d} + q^*\widehat{\text{se}}]. \tag{55}$$

The $0.025$ tail level is the Bonferroni allocation for two editors. Conditional on the fitted parameters, asymptotic validity requires independent content clusters, finite moments of order greater than two, and nonzero variance. This centered cluster-level wild bootstrap does not require exact sign symmetry of the raw $d_i$ and is not a design-exact randomization test.

## H Experimental Protocol

### H.1 Model, data, and prompt construction

The frozen backbone is Gemma 2 2B instruction tuned (Gemma Team et al., 2024). Content comes from the sentence-transformers COCO captions dataset, whose underlying images and captions derive from Microsoft COCO (Lin et al., 2014). The source table stores five captions for each image in consecutive rows. Both studies select one caption from each five-row image group before splitting, and no selected image group appears twice. Training, calibration, and test contents are therefore disjoint at the image-group level.

Both instructions use the same response scaffold and differ only in style. The templates are:

```
Task: Continue the description while preserving every
entity, count, action, and spatial relation.
Style: literal.
Scene: {content}
Response: In this scene, the description begins
```

for the source and the same text with `Style: vivid fantasy.` for the target. This construction makes content identity explicit. It does not guarantee semantic preservation, which remains an empirical diagnostic.

### H.2 Editors and fitting

AcT transports activation distributions with optimal-transport-guided maps (Rodriguez et al., 2025a). LinEAS learns multi-layer activation maps with a downstream distributional loss (Rodriguez et al., 2025b). We fit both candidate editors with the public implementations from their repositories and the fantasy corpus distributed with LinEAS. This corpus is separate from all COCO contents used to fit or evaluate PCRT.

Both candidates use the same 48 source and 48 target training texts, all 52 post-attention and post-feedforward normalization sites across 26 layers, every token position, and strength 1. AcT fits one repository `LinearOTHook` per site in its at-once, all-quantile configuration. LinEAS initializes

one repository `BPLinearHook` per site at identity and minimizes the mean per-neuron sorted empirical squared transport loss. Its optimizer is Adam with learning rate $10^{-3}$, a cosine schedule ending at $10^{-6}$, batch size 48, and a maximum of 400 steps.

Sixteen source and 16 target texts form a fixed calibration set checked every 25 steps under a predeclared stopping rule. After at least 200 steps, five consecutive checks with relative improvement below $10^{-4}$ trigger stopping; optimization stops at step 325. Neither candidate uses COCO calibration or test results for model selection; after fitting, PCRT receives only the candidate's final normalized state and learns independently from paired COCO target residuals.

Each study and editor fits its own $U$, numerical rank, shrinkage intensity, and linear conditional predictor from training data. The fixed calibration split verifies numerical validity and directional agreement; all model, feature, and decision choices remain fixed before test evaluation.

### H.3 TWO PROBABILITY EVALUATIONS

The common-history evaluation uses a $128/64/64$ training, calibration, and test split on its own 256-content cohort. It compares five positions under one shared response scaffold. The fitted residual rank is 640 for both editors, and the controller uses the explicit causal-prefix feature in Section E. Candidate, target, and repaired states are compared for the same content and response history. This study isolates the current output-kernel correction.

The target-path evaluation uses a disjoint $64/32/64$ split on a 160-content cohort and 16 response positions. The fitted residual rank is 1024 for both editors, and the controller uses the pooled current-state feature. There is one target path per training and calibration content and two independent target paths per test content. Each target token is paired with the state that produced its sampling kernel. Candidate states are then collected under AcT or LinEAS on the identical target prefix, and the repaired state is computed from the candidate state. The two studies therefore share the method class and probability target while retaining separate data, bases, and controllers.

### H.4 REFERENCE CONTROLLER SIZE AND COMPUTATION

The target-path controller has hidden width $d = 2304$ and fitted rank $r = 1024$. Its deployed 32-bit floating-point (FP32) arrays are $U$, the precomputed product $GU$, the $r \times r$ coefficient $B$, and two length-$r$ mean vectors. Per editor, this is

$$2dr + r^2 + 2r = 5{,}769{,}216 \tag{56}$$

stored values, or 22.0 mebibytes (MiB). The output matrix belongs to the frozen model and is not counted again. One repair evaluates two $d \times r$ products and one $r \times r$ product, requiring about $5.77$ million multiply-adds per token plus vector operations. It does not run the target prompt or target branch at deployment.

Fitting is offline. The target-path study forms $64 \times 16 = 1024$ training pairs, computes the training residual subspace from a $1024 \times 2304$ matrix, and solves the analytic shrinkage system in a $1024$-sample dual. These operation counts describe the Gemma 2 reference implementation. Appendix L.5 reports measured latency, throughput, controller size, and fitting time for the external backbones.

### H.5 DEPLOYED NUMERICAL KERNEL

The Gemma 2 target-path studies score the deployed probability kernel $\widetilde{p}$ from Equation 3, using Brain Floating Point 16 (BF16). For arm $A \in \{T, C, R\}$,

$$\widetilde{a}_t^A = \mathrm{head}_{\mathrm{BF16}}(x_t^A), \tag{57}$$

$$q_t^A = \mathrm{BF16}(\widetilde{a}_t^A/30), \qquad v_t^A = \mathrm{BF16}(\tanh(q_t^A)), \tag{58}$$

$$\widetilde{\ell}_t^A = \mathrm{BF16}(30 v_t^A). \tag{59}$$

Here $\mathrm{head}_{\mathrm{BF16}}$ denotes the vocabulary-head numerical primitive, including matrix-multiplication rounding. The FP32 controller correction is cast to BF16 before addition to the candidate: the deployed repaired state is $\mathrm{BF16}(x_t^C + \mathrm{BF16}(U\widehat{m}(\phi_t)))$, with the inner correction evaluated in FP32.

Equation 16 describes its continuous counterpart. The deployed logits $\widetilde{\ell}_t^A$ are then cast to FP32 for log-softmax, probabilities, and KL. Target sampling and all scored target-path arms use this law. Incremental cached execution supplies the primary states. A bounded full-prefix comparison checks this implementation because distinct low-precision schedules need not be bit-identical. The external Qwen and Llama evaluations retain each model's native linear output transformation and the same BF16-to-FP32 probability evaluation protocol.

The common-history diagnostic uses a separate, shared scoring path for all arms. It casts the state to BF16, applies the BF16 vocabulary head, and casts its output to FP32 before evaluating $30\tanh(a/30)$, log-softmax, KL, and total variation. Thus its full-vocabulary post-softcap KL isolates repair on fixed histories; it does not use the native BF16 softcap of the sequence studies. Its output-head error is the continuous quadratic form $\|W(x^T - x^A)\|_2^2$, computed from stored states.

### H.6 Cross-backbone and public-task protocol

The cross-backbone study changes the model while keeping the caption transformation and estimator fixed. It evaluates Qwen 2.5 7B Instruct, Qwen 3 4B Instruct, and Llama 3.1 8B Instruct (Qwen et al., 2024; Yang et al., 2025; Grattafiori et al., 2024). The frozen repositories and revisions are:

- `Qwen/Qwen2.5-7B-Instruct`, revision `a09a35458c702b33eeacc393d103063234e8bc28`.
- `Qwen/Qwen3-4B-Instruct-2507`, revision `cdbee75f17c01a7cc42f958dc650907174af0554`.
- the public mirror `NousResearch/Meta-Llama-3.1-8B-Instruct`, revision `d10aef7999a2b5ba950ab3974312feeedbfe0b77`.

For each model, 64 paired COCO contents fit AcT at all decoder blocks with strength one. Separate disjoint cohorts of 64, 32, and 64 contents fit, check, and test PCRT. Each test content contributes two target-generated paths of 32 positions. The post-editor arm corrects the AcT candidate, while the direct-source arm applies the same estimator class to the unedited source state. Model-specific output matrices define $G$, and every basis and controller is fitted separately within its model and arm.

The public-task study changes both the content domain and the measured behavior. It uses Qwen 2.5 7B Instruct and five single-constraint categories from IFEval: no comma, quoted response, lowercase response, title format, and specified ending (Zhou et al., 2023). For each category, the first 20 unique augmented examples whose keys do not occur in the formal test set form the training split. The formal test contributes 16, 14, 13, 13, and 17 items, respectively, for 73 independent test items. The source text is the task with its formatting constraint removed, and the paired target is the original constrained prompt. Each test item contributes two target paths of 32 positions for sequence KL and one greedy generation of at most 128 tokens for the official strict evaluator. Reported macro quantities first average within category and then weight the five categories equally.

The matched paired-prompt comparison uses the same training and test items. We fit the published all-layer Prompt Steering Replacement model with its mean-squared-error objective, denoted A-PSR-MSE, at all decoder blocks (Heyman & Vandeputte, 2026). Each category is fitted for 15 epochs with batch size one, learning rate $10^{-3}$, weight decay $10^{-6}$, and three independent training seeds. PCRT and A-PSR-MSE are evaluated as separate direct-control arms from the same source prompts. No target instruction is available to either deployed controller.

### H.7 Primary tests and decision rules

The common-history analysis reports head error and diagnostic post-softcap KL for each editor. Its main uncertainty summaries are paired percentile-bootstrap intervals from 20,000 resamples of the 64 independent content clusters. The predeclared 10,000 one-sided sign-flip calculations at family threshold $0.05/4$ are retained as sensitivity checks. Exact sign-flip inference would require randomized within-pair label exchange or a sign-symmetric paired-difference null, neither of which follows from deterministic content pairing. The target-path analysis has one mean sequence improvement per editor. A Bonferroni simultaneous interval entirely above zero indicates positive mean improve-

ment, an interval below zero indicates negative mean improvement, and an interval containing zero is inconclusive.

The two external COCO families use 10,000 centered studentized Rademacher draws with a maximum absolute statistic across the three backbones. IFEval first averages within each prespecified category and then weights the five categories equally. Its distributional family compares PCRT with the source and with the A-PSR-MSE seed mean under the same maximum-statistic construction. Strict compliance uses 10,000 stratified paired item-bootstrap draws that preserve category counts.

### H.8 CONTROLS

The control suite separates the proposed mechanism into testable components:

1. The unrepaired candidate measures the error left by the existing editor.
2. The paired residual mean measures the gain from a global same-content correction without candidate dependence.
3. The matched different-content residual control preserves the donor residual multiset and fits the same predictor class after balancing prompt length, response position, candidate entropy, and output-head residual norm.
4. The cross-content target-state swap pairs each candidate with another content's target state. It changes both the donor residual and its candidate reference, so it serves as a broad pairing stress control rather than an isolated residual swap.
5. The same-law identity control tests whether the implementation introduces a change when candidate and target laws agree.
6. Calibration repeats the directional measurement on held-out, non-test content without changing the fitted method.
7. The matched geometry ablation replaces $G$ with $I$ during common-history fitting, while retaining the same data, causal feature, rank rule, shrinkage estimator, and $G$-based test error.
8. The temporal rank-covariance control treats each hidden coordinate separately. It rank-normalizes that coordinate at every position, applies a per-coordinate lower-triangular map fitted to the candidate and target shrinkage covariance estimates, and decodes the transformed scores to the empirical candidate marginal scale. It changes second-order rank dependence without claiming to match the full joint law, uses no same-content residual, and reads no future position.
9. The direct-transport control treats the literal source state as the candidate and fits a separate controller from the same $G$-based estimator class toward the paired target state.

## I DETAILED NUMERICAL RESULTS

### I.1 COMMON-HISTORY PRIMARY QUANTITIES

Both editors show large common-history reductions in output-head error and full-vocabulary post-softcap KL, with at least 63 of 64 test contents improving on each metric (Table 7). The output-head error is $\|W(x^T - x^A)\|_2^2$, averaged within content before inference. KL uses the common-history scoring path above. Reported values are rounded for readability; generated arrays retain full numerical precision.

The mean improvement intervals from an independent content-cluster bootstrap are $[705{,}527, 803{,}082]$ and $[0.499, 0.580]$ for AcT head error and KL. The corresponding LinEAS intervals are $[567{,}772, 697{,}307]$ and $[0.419, 0.483]$. These 95% percentile intervals summarize mean-reduction magnitude. The planned sign-flip sensitivity calculations each have an unadjusted Monte Carlo tail fraction of $1/10001$. They are not interpreted as design-exact randomization tests.

Table 7: Complete common-history primary results. Positive counts test contents with lower error after repair.

| | Output-head error | | | Exact post-softcap KL | | |
|---|---|---|---|---|---|---|
| Editor | Candidate | Repaired | Positive | Candidate | Repaired | Positive |
| AcT | 901,805 | 147,216 | 64/64 | 0.762 | 0.221 | 63/64 |
| LinEAS | 793,922 | 162,641 | 64/64 | 0.681 | 0.230 | 64/64 |

### I.2 Common-history controls

Neither cross-content target-state swapping nor the temporal rank-covariance control reproduces the common-history gain (Table 8).

Table 8: Common-history exact post-softcap KL controls. Lower is better.

| Editor | Candidate | Conditional repair | Target-state swap | Temporal rank-covariance |
|---|---|---|---|---|
| AcT | 0.762 | 0.221 | 0.772 | 0.800 |
| LinEAS | 0.681 | 0.230 | 0.715 | 0.693 |

The corresponding repaired, target-state-swap, and temporal rank-covariance output-head errors are 147,216, 502,026, and 912,304 for AcT. They are 162,641, 489,193, and 802,513 for LinEAS. The target-state-swap and temporal rank-covariance controls are worse than the candidate on exact KL for both editors. The same-law control is exactly unchanged.

Table 9: Matched common-history geometry ablation and direct transport. All rows are evaluated with output-head error $G$. Positive counts test contents with lower error after repair.

| Candidate | Fit geometry | Before | After | Removed | Positive |
|---|---|---|---|---|---|
| AcT | $G$ | 901,805 | 147,203 | 83.68% | 64/64 |
| AcT | $I$ | 901,805 | 146,811 | 83.72% | 63/64 |
| LinEAS | $G$ | 793,922 | 162,700 | 79.51% | 64/64 |
| LinEAS | $I$ | 793,922 | 163,587 | 79.40% | 64/64 |
| Source | $G$ | 694,230 | 123,598 | 82.20% | 64/64 |

For AcT, the Euclidean-fit error minus the matched $G$-fit replay is $-392$, with paired bootstrap interval $[-6{,}797, 6{,}945]$. The LinEAS difference is 887, with interval $[-5{,}849, 8{,}408]$. The $G$ rows use the same offline numerical path as the Euclidean rows. Their means differ from Table 7 by less than $0.1\%$. The direct source-to-target removed fraction has interval $[79.6\%, 84.6\%]$. These held-out controls use the complete common-history data and the same analytic estimator. The direct row measures output-head error only. It is not a target-path or decoded-text comparison.

### I.3 Target-path primary quantities

Sequence repair improves every test content for both editors, as the full distribution of content-level gains shows (Table 10).

The Bonferroni simultaneous intervals are $[7.162, 8.352]$ for AcT and $[3.205, 4.000]$ for LinEAS. Both unadjusted two-sided finite Monte Carlo tail probabilities are $1/10001$. The intervals provide the inferential conclusion. The finite-resolution probabilities are supporting summaries.

### I.4 Secondary distribution quantities

Total variation and pre-softcap logit error decrease in the same direction as the primary sequence-KL result (Table 11). Here logit error is $\|\widetilde{a}_t^T - \widetilde{a}_t^A\|_2^2$, using the BF16 vocabulary-head outputs. It differs from the continuous output-head quadratic form by numerical rounding.

Table 10: Complete content-level target-path sequence results. Std. denotes standard deviation. Positive counts test contents with lower sequence KL after repair.

| Editor | Candidate | Repaired | Mean gain | Std. | Range | Positive |
|---|---|---|---|---|---|---|
| AcT | 14.260 | 6.503 | 7.757 | 2.091 | [3.484, 14.191] | 64/64 |
| LinEAS | 10.080 | 6.478 | 3.603 | 1.395 | [0.668, 7.611] | 64/64 |

Table 11: Secondary target-path distribution quantities, summed over 16 positions.

| | Total variation sum | | | Pre-softcap logit error sum (millions) | | |
|---|---|---|---|---|---|---|
| Editor | Candidate | Repaired | Reduction | Candidate | Repaired | Reduction |
| AcT | 6.597 | 4.459 | 32.4% | 14.070 | 6.799 | 51.7% |
| LinEAS | 5.585 | 4.394 | 21.3% | 7.683 | 6.688 | 12.9% |

These secondary quantities agree with the sequence KL direction. Their relative reductions differ because the metrics weight vocabulary errors differently. We therefore keep sequence KL as the primary criterion.

### I.5 Sequence controls and calibration

The initial target-path controls separate candidate dependence from a broad disruption of candidate-to-target pairing (Table 12). The next subsection isolates residual pairing with a matched different-content donor.

Table 12: Target sequence KL for the primary method and initial controls. The target-state swap uses another content's complete target state.

| Editor | Candidate | PCRT | Paired mean | Target-state swap | Same law |
|---|---|---|---|---|---|
| AcT | 14.260 | 6.503 | 8.084 | 69.917 | 0 |
| LinEAS | 10.080 | 6.478 | 7.552 | 72.219 | 0 |

On calibration content, AcT moves from $13.884$ to $6.997$, and LinEAS moves from $10.011$ to $6.416$. Calibration therefore agrees in direction without being used to select the method.

### I.6 Matched different-content residual control

The matched residual control tests whether correct pairing contributes information beyond observed prompt and output statistics. For editor $e$, training content $i$, and position $t$, define the genuine residual

$$d^e_{i,t} = x^T_{i,t} - x^{C,e}_{i,t}. \tag{60}$$

The control replaces $d^e_{i,t}$ with $d^e_{\pi(i),t}$ while retaining the feature from $x^{C,e}_{i,t}$. The same no-fixed-point bijection $\pi$ is shared across AcT, LinEAS, and all 16 positions. It therefore preserves the complete donor residual multiset for each editor and keeps response position exact.

The donor assignment is fitted on training data without test outcomes. Let $r^L_i$ contain the source and target prompt-length ranks, $r^H_{e,i,t}$ the candidate-entropy rank, and $r^R_{e,i,t}$ the rank of the output-head residual norm $\|W d^e_{i,t}\|_2$. Ranks use average ties and are scaled to $[0, 1]$. The assignment minimizes the total cost under $\pi(i) \neq i$, where

$$C_{ij} = \frac{1}{3}\left[\frac{1}{2}\|r^L_i - r^L_j\|^2_2 + \frac{1}{2L}\sum_{e,t}(r^H_{e,i,t} - r^H_{e,j,t})^2 + \frac{1}{2L}\sum_{e,t}(r^R_{e,i,t} - r^R_{e,j,t})^2\right]. \tag{61}$$

The matched assignment lowers mean cost from $0.1638$ for a fixed random derangement to $0.0855$, a $47.8\%$ reduction. Table 13 reports the component balance.

Table 13: Training-only donor balance for the matched residual control. Entries are mean absolute rank gaps; lower values indicate closer donor pairs.

| Matched statistic | Matched donor | Random donor |
|---|---|---|
| Source and target prompt length | 0.0878 | 0.3033 |
| Candidate entropy | 0.2758 | 0.3440 |
| Output-head residual norm | 0.2770 | 0.3352 |

The test comparison keeps the model, target paths, residual basis, estimator class, and probability kernel fixed. Each editor refits only the linear coefficients from the matched donor responses. For content $i$, the primary contrast is

$$Q_i^e = \widehat{\mathrm{KL}}^{\,e}_{\text{matched residual},i} - \widehat{\mathrm{KL}}^{\,e}_{\text{PCRT},i}. \tag{62}$$

Each value averages two target paths over 16 positions. We apply 10,000 centered studentized Rademacher draws to the 64 independent content contrasts and allocate level $0.025$ to each editor.

Table 14: Matched different-content residual results. Lower sequence KL is better. Intervals are Bonferroni simultaneous intervals for matched residual minus PCRT; positive counts indicate contents on which PCRT has lower KL.

| Editor | PCRT | Matched | Random | Gap | Gap interval | Positive |
|---|---|---|---|---|---|---|
| AcT | 6.503 | 11.093 | 11.377 | 4.590 | $[4.246, 4.934]$ | $64/64$ |
| LinEAS | 6.478 | 9.947 | 10.220 | 3.469 | $[3.116, 3.822]$ | $64/64$ |

For both editors, the unadjusted two-sided finite Monte Carlo tail probability is $1/10001$; the simultaneous intervals determine the decision.

Figure 5 places the matched residual result beside the paired residual mean. The matched control uses candidate-dependent prediction with incorrect content pairing, while the paired mean uses correct pairing without candidate dependence. PCRT outperforms both controls for both editors.

The matched residual control supports a specific mechanism claim. Under the tested Gemma 2, COCO, AcT, and LinEAS protocol, correct pairing remains predictive after exact response-position matching and reduced training-only rank imbalance in prompt length, candidate entropy, and output-head residual norm. This control neither defines general semantic similarity nor establishes that all nuisance differences were removed.

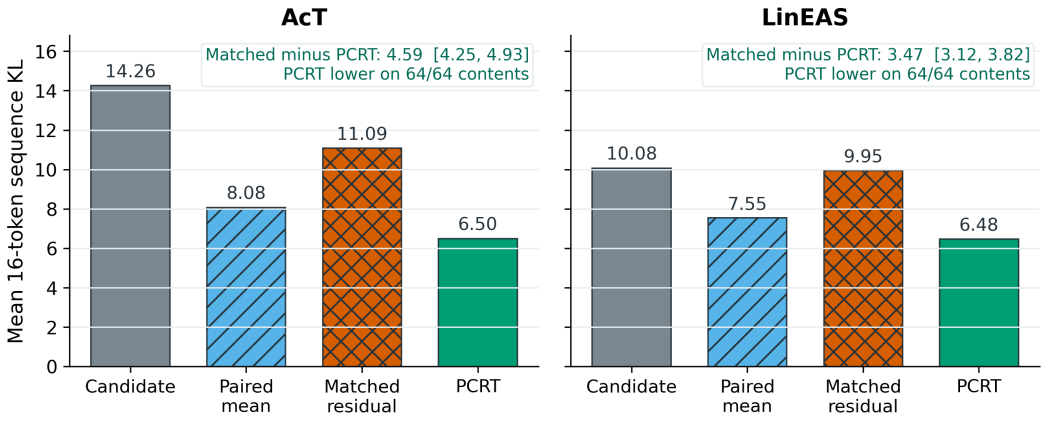


Figure 5: Correct pairing and candidate conditioning both contribute to the primary sequence gain. The matched residual control retains candidate-dependent fitting but replaces each training residual with a different-content residual selected by training-only minimum-cost rank matching. The paired mean keeps correct pairing without candidate dependence. Intervals inside the panels are simultaneous 95% intervals for matched residual minus PCRT. Lower sequence KL is better.

# J Stability Analyses

## J.1 Position-level effects

Mean conditional KL improvement remains positive at all 16 response positions for both editors (Table 15).

Table 15: Mean candidate-minus-repaired KL at each target-path position.

| Position | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| AcT | .384 | .389 | .998 | .525 | .536 | .479 | .501 | .483 |
| LinEAS | .600 | .251 | .555 | .167 | .204 | .203 | .172 | .195 |
| Position | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| AcT | .429 | .303 | .421 | .450 | .408 | .476 | .453 | .523 |
| LinEAS | .126 | .147 | .112 | .168 | .146 | .165 | .191 | .202 |

The position curves do not show a decay toward zero over the measured horizon. AcT has its largest mean improvement at position 3, while LinEAS has larger early gains followed by smaller positive gains. The experiment does not extrapolate this pattern beyond 16 positions.

## J.2 Path-level and content-level signs

AcT has positive improvement on all 128 individual target paths. LinEAS has positive improvement on 125 of 128 paths. The three negative LinEAS paths occur within contents whose predeclared two-path average remains positive. This distinction matters because content, rather than a sampled path, is the independent unit.

The minimum content-average improvement is $3.484$ for AcT and $0.668$ for LinEAS. Thus even the smallest observed content-average gain is positive for both editors. The simultaneous interval concerns the population mean and does not claim a uniform lower bound for every future content.

## J.3 Leave-one-content-out means

Deleting each test content in turn gives AcT mean improvements in $[7.655, 7.825]$ and LinEAS means in $[3.539, 3.649]$. No single content controls either primary mean. This analysis was con-

ducted after the fixed primary inference and serves as a stability description rather than a new hypothesis test.

# K PREREGISTERED CROSS-ATTRIBUTE CONFIRMATION

## K.1 QUESTION AND STUDY DESIGN

The cross-attribute experiment tests whether the primary sequence result depends on the vivid-fantasy target. It keeps the frozen Gemma 2 2B model, COCO domain, deployed probability kernel, 16-position horizon, two target paths per test content, controller family, and content-level inference unchanged. The source requests a neutral factual continuation. The new target requests a warm, positive, and appreciative continuation while remaining factual. This study changes the target attribute while retaining the same controller and evaluation protocol.

The editor and controller use fully disjoint content cohorts. The official AcT implementation is fitted on 64 paired contents for the new attribute. PCRT then uses 64 training, 32 calibration, and 64 test contents, none of which appears in the editor-fit cohort. The test has 128 target paths, and all fitted quantities are fixed before test evaluation. Inference uses a centered studentized Rademacher wild-bootstrap interval over 64 content-level sequence-KL improvements, with a prespecified positive-result criterion that requires the lower endpoint to exceed zero. No attribute, layer, rank, dose, horizon, estimator, threshold, or test subset is selected from the test result.

## K.2 PRIMARY SEQUENCE RESULT

The mean effect reproduces under the preregistered criterion (Table 16). The 16-token sequence KL falls from $3.005$ to $2.201$, which removes $26.8\%$ of the candidate error. The mean improvement is $0.804$, with standard error $0.074$, interval $[0.656, 0.952]$, and two-sided finite Monte Carlo tail probability $1/10001$. The simultaneous interval provides the inferential conclusion.

Table 16: Preregistered warm-appreciative target-path confirmation for AcT. Lower sequence KL is better, the improvement interval is simultaneous $95\%$, and positive counts use the two-path content average.

| Editor | Candidate | PCRT | Reduction | Improvement interval | Positive |
|---|---|---|---|---|---|
| AcT | 3.005 | 2.201 | 26.8% | [0.656, 0.952] | 59/64 |

The result is positive at each aggregation level used for stability description. The median content improvement is $0.748$, with range $[-0.647, 2.390]$. In total, 59 of 64 contents and 115 of 128 individual paths improve. Five contents worsen, so the result is a positive population-mean statement rather than a uniform guarantee. Mean improvement is positive at all 16 positions, and leave-one-content-out means remain in $[0.779, 0.827]$.

Table 17: Mean candidate-minus-repaired conditional KL at each position for the cross-attribute confirmation.

| Position | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| Gain | .109 | .080 | .132 | .069 | .030 | .038 | .026 | .036 |
| Position | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| Gain | .039 | .037 | .041 | .032 | .016 | .036 | .048 | .034 |

## K.3 CONTROLS AND SUPPORTING MEASUREMENTS

The controls distinguish external confirmation from a repeated mechanism claim. A cross-content target-state swap raises sequence KL to $70.510$, and the same-law identity error is exactly zero. The paired residual mean reaches $2.171$, which is numerically close to and slightly below the full controller at $2.201$. Their paired descriptive difference is $-0.030$ with standard error $0.046$. This

comparison was not a separate preregistered test. It shows that the cross-attribute result establishes same-content repair but does not independently establish an advantage from candidate conditioning.

Table 18: Cross-attribute primary and control quantities summed over 16 target-path positions. Lower is better.

| Arm | Sequence KL | Total variation | Pre-softcap logit error (millions) |
|---|---|---|---|
| AcT candidate | 3.005 | 3.054 | 3.152 |
| PCRT | 2.201 | 2.591 | 2.673 |
| Paired residual mean | 2.171 | 2.652 | 2.205 |
| Target-state swap | 70.510 | 12.108 | 47.773 |

The secondary quantities agree with the primary direction. PCRT reduces the total-variation sum by $15.2\%$ and pre-softcap logit error by $15.2\%$. Calibration sequence KL moves from $3.088$ to $2.214$ without being used for selection. The same deployed probability kernel evaluates target, candidate, and repaired states on matched prefixes.

Greedy generations provide a supporting behavior check. The target, AcT candidate, and repaired outputs have warm-appreciative lexicon rates of $0.281$, $0.688$, and $0.500$. Source-caption alphabetic-unigram recall is $0.698$, $0.510$, and $0.598$, respectively. Paired-target unigram Jaccard is $1.000$, $0.316$, and $0.358$. The repaired output moves toward the target on these diagnostics, which remain separate from the formal decision.

### K.4 Interpretation across attributes

This study establishes same-model, same-domain cross-attribute transfer for one AcT editor. Under a distinct preregistered target instruction and fresh content cohorts, PCRT reduces AcT target-sequence KL under the same deployment and inference protocol. The primary literal-to-fantasy task isolates the added value of candidate conditioning, while the second attribute independently confirms same-content repair. Cross-model evidence for the literal-to-fantasy task appears in Appendix L.

## L Cross-Backbone and Public-Task Evaluation

### L.1 Post-editor correction across backbones

The AcT correction reproduces on every external backbone. Table 19 reports the complete 32-token sequence results. The simultaneous intervals are for the absolute mean sequence-KL gain, and all lower endpoints exceed zero. Every one of the 64 shared test contents improves on each backbone.

### L.2 Direct source correction across backbones

The same estimator class also corrects the unedited source across all three external backbones. Table 20 shows reductions from $45.8\%$ to $71.8\%$. All three simultaneous intervals exclude zero. Qwen 2.5 and Qwen 3 improve on every test content, while Llama 3.1 improves on 62 of 64.

### L.3 Public-task distribution repair and compliance

The public-task result connects probability matching to an official behavioral measure. Table 21 gives every frozen category. With equal weight per category, source sequence KL falls from $39.017$ to $30.367$, and strict compliance rises from $7.9\%$ to $52.0\%$. The macro KL gain is $8.650$ with simultaneous interval $[5.388, 11.912]$. The macro compliance gain is $44.1$ percentage points with stratified paired-bootstrap interval $[35.5, 52.5]$.

### L.4 Matched paired-prompt comparison

Matched PSR is stronger for direct prompt replacement under this protocol. The comparison evaluates PCRT and A-PSR-MSE on the same 100 training items and 73 formal test items; Tables 22

Table 19: AcT post-editor correction across three additional backbones. Lower sequence KL is better. Intervals report the simultaneous 95% interval for the absolute mean gain; positive counts test contents with lower sequence KL after correction.

| Backbone | AcT | AcT + PCRT | Reduction | Gain interval | Positive |
|---|---|---|---|---|---|
| Qwen 2.5 7B | 127.944 | 38.919 | 69.6% | [85.116, 92.935] | 64/64 |
| Qwen 3 4B | 203.657 | 53.029 | 74.0% | [146.004, 155.253] | 64/64 |
| Llama 3.1 8B | 177.316 | 45.821 | 74.2% | [125.205, 137.785] | 64/64 |

Table 20: Direct source correction across three additional backbones. Lower sequence KL is better. Intervals report the simultaneous 95% interval for the absolute mean gain; positive counts test contents with lower sequence KL after correction.

| Backbone | Source | Source + PCRT | Reduction | Gain interval | Positive |
|---|---|---|---|---|---|
| Qwen 2.5 7B | 53.668 | 16.498 | 69.3% | [33.270, 41.069] | 64/64 |
| Qwen 3 4B | 101.123 | 28.525 | 71.8% | [67.090, 78.106] | 64/64 |
| Llama 3.1 8B | 25.897 | 14.026 | 45.8% | [9.935, 13.808] | 62/64 |

and 23 report all three fitted A-PSR-MSE seeds. Its seed-mean macro sequence KL is 9.329, compared with 30.367 for PCRT, and it has lower item-level sequence KL on all 73 items. The PCRT gain relative to A-PSR-MSE is $-21.038$ with simultaneous interval $[-23.271, -18.805]$. Its seed-mean macro compliance is 65.8%, compared with 52.0% for PCRT; the difference is $-13.8$ percentage points with interval $[-22.1, -5.8]$. The post-editor and source-correction results establish the distinct repair capability claimed for PCRT.

### L.5 EFFICIENCY

PCRT adds a small online latency cost to AcT across the external backbones. Table 24 reports incremental offline cost, batch-one latency, and capacity throughput. The controller adds 0.6% to 1.7% mean latency per generated token, and offline fitting takes about one second after paired states are collected. Parameter and storage columns count only the added PCRT controller. Each throughput arm uses its own largest verified batch, so the final two columns describe capacity rather than matched-batch speed.

Table 21: Direct source correction on five prespecified IFEval categories. Compliance is the official strict score. The macro row weights categories equally; 73 is the total item count.

| Category | $n$ | Source KL | PCRT KL | Source compliance | PCRT compliance |
|---|---|---|---|---|---|
| No comma | 16 | 74.610 | 58.216 | 18.8% | 93.8% |
| Quoted response | 14 | 33.393 | 18.324 | 7.1% | 14.3% |
| Lowercase response | 13 | 41.344 | 34.989 | 0.0% | 53.8% |
| Title format | 13 | 37.862 | 26.475 | 7.7% | 92.3% |
| Specified ending | 17 | 7.875 | 13.830 | 5.9% | 5.9% |
| Equal-category macro | 73 total | 39.017 | 30.367 | 7.9% | 52.0% |

Table 22: Matched-data A-PSR-MSE sequence KL by training seed. Lower is better. The macro row weights categories equally.

| Category | PCRT | A-PSR-MSE seed 1 | Seed 2 | Seed 3 |
|---|---|---|---|---|
| No comma | 58.216 | 13.114 | 12.836 | 12.987 |
| Quoted response | 18.324 | 6.166 | 6.992 | 6.696 |
| Lowercase response | 34.989 | 9.451 | 9.166 | 9.367 |
| Title format | 26.475 | 12.394 | 13.161 | 13.014 |
| Specified ending | 13.830 | 4.931 | 4.980 | 4.679 |
| Equal-category macro | 30.367 | 9.211 | 9.427 | 9.349 |

Table 23: Matched-data A-PSR-MSE strict compliance by training seed. Higher is better. The macro row weights categories equally.

| Category | PCRT | A-PSR-MSE seed 1 | Seed 2 | Seed 3 |
|---|---|---|---|---|
| No comma | 93.8% | 100.0% | 93.8% | 100.0% |
| Quoted response | 14.3% | 35.7% | 28.6% | 28.6% |
| Lowercase response | 53.8% | 92.3% | 84.6% | 100.0% |
| Title format | 92.3% | 100.0% | 100.0% | 100.0% |
| Specified ending | 5.9% | 5.9% | 5.9% | 11.8% |
| Equal-category macro | 52.0% | 66.8% | 62.6% | 68.1% |

Table 24: Incremental cost of adding PCRT to AcT. Arrows compare AcT with AcT plus PCRT. Batch-one latency averages 128 measured steps after 16 warmup steps. Each arm uses its own largest verified batch for the capacity-throughput measurement.

| Backbone | Fit $\Delta$ (s) | Params $\Delta$ (M) | Size $\Delta$ (MiB) | Latency (ms/token) | Increase | Capacity batch | Capacity token/s |
|---|---|---|---|---|---|---|---|
| Qwen 2.5 7B | 1.1 | 18.9 | 108.3 | 31.0 $\rightarrow$ 31.5 | 1.7% | 3578 $\rightarrow$ 3357 | 3712 $\rightarrow$ 3642 |
| Qwen 3 4B | 1.0 | 14.2 | 80.9 | 43.6 $\rightarrow$ 44.1 | 1.2% | 2101 $\rightarrow$ 2053 | 3015 $\rightarrow$ 2988 |
| Llama 3.1 8B | 1.2 | 21.0 | 120.3 | 34.6 $\rightarrow$ 34.8 | 0.6% | 2057 $\rightarrow$ 1971 | 2956 $\rightarrow$ 2951 |

# M GENERATED-TEXT DIAGNOSTICS

Generated text provides an interpretable check on the probability result. It remains secondary because greedy decoding maps a full distribution to one output and lexical metrics are sensitive to wording.

Let $\mathcal{A}(s)$ be the set of lowercase alphabetic unigrams matched by `[a-z]+` in string $s$. Source recall is $|\mathcal{A}(c) \cap \mathcal{A}(g)| / \max(|\mathcal{A}(c)|, 1)$ for source caption $c$ and generation $g$. Target Jaccard uses $|\mathcal{A}(t) \cap \mathcal{A}(g)|$ in the numerator and $\max(|\mathcal{A}(t) \cup \mathcal{A}(g)|, 1)$ in the denominator for paired target generation $t$. Unique texts counts distinct decoded strings among the 64 outputs.

Table 25: Thirty-two-token greedy generation diagnostics. Source recall is source-caption alphabetic-unigram recall; target Jaccard uses the paired target generation. Higher is better for all columns.

| Arm | Source recall | Target Jaccard | Unique texts |
|---|---|---|---|
| AcT candidate | 0.266 | 0.142 | 64/64 |
| AcT repaired | 0.411 | 0.238 | 64/64 |
| LinEAS candidate | 0.321 | 0.158 | 64/64 |
| LinEAS repaired | 0.434 | 0.256 | 64/64 |

The rate of outputs containing a predeclared fantasy lexicon is $7.8\%$ for the target, $42.2\%$ before and $18.8\%$ after AcT repair, and $31.3\%$ before and $7.8\%$ after LinEAS repair. Lower rates move toward the paired target in this diagnostic. The lexicon measures a narrow lexical property rather than overall fantasy quality.

## M.1 REPRESENTATIVE HELD-OUT TEXT IN FIGURE 1

The text excerpts use the content "A group of people are each holding umbrellas on a rainy day." We averaged the two editor improvement ranks, selected the item closest to the median joint rank, and broke ties by dataset order. This deterministic rule was applied before reading the associated generations. Its content-level sequence improvements are $7.790$ for AcT and $3.482$ for LinEAS.

For AcT, the candidate describes "the celestial dance of the heavens" without mentioning rain or umbrellas. The repaired output returns to "a symphony of rain" and "The umbrellas". The LinEAS candidate mentions "shimmering rain", while its repaired output begins with "a symphony of raindrops", matching the target opening. The example illustrates how correction can move generated text toward its same-content target; the aggregate conclusion uses all 64 test contents.

## M.2 INTERPRETATION

Both repaired arms improve source-caption recall and paired-target Jaccard in this task. All candidate and repaired arms produce 64 distinct outputs on the 64 test contents. These lexical diagnostics provide supporting evidence about content and diversity. The target-path full-vocabulary KL remains the direct measure of the chosen target distribution, while exact semantic preservation, factuality, safety, and general capability require dedicated evaluations.

# N REPRODUCING THE RESULTS

## N.1 RESULT FILES AND FIGURE RECONSTRUCTION

The Supplementary Material contains the experimental program and figure-generation scripts. Running the program produces measurements for the common-history and target-path studies, the matched different-content residual control, the cross-attribute confirmation, external backbones, and IFEval. The plotting scripts read these outputs to rebuild Figures 1, 2, 3, 4, and 5, and record the plotted aggregates. The probability curves and method drawing in Figure 1 are schematic.

All reported means, content-level or item-level sequence differences, simultaneous intervals, and stratified compliance intervals come from one end-to-end execution of the released reproduction

script. Its result package contains the paired measurements and corresponding statistical summaries, so each table and plot can be traced to the same fitted editors, controllers, and generated paths.

### N.2 Numerical implementation checks

The numerical checks preserve the probability interface used by each scientific estimand. Candidate, target, and repaired states use the same full-vocabulary scoring path within each study. For target-path evaluation, the native output transformation reproduces the model's scored logits within numerical tolerance. Target sampling reuses the state that produced each sampled token, and incremental batches follow the same position rule as generation. Equal target and candidate laws leave the output unchanged. These tolerances check implementation conformance and do not enter statistical inference.

### N.3 Scope of the released evidence

The supplementary code supports a complete rerun from public resources and aggregate reconstruction from the resulting measurements. The single-cell reproduction script downloads the listed model weights, repository revisions, and raw datasets; fits every editor and controller; and computes the main and appendix measurements. It does not consume earlier fitted models or experiment results. The generated compact arrays permit later table and figure reconstruction without loading the language models again.

## O Scope, Limitations, and Extensions

### O.1 What the experiment establishes

The evidence links output geometry, same-content residual prediction, and sequence-level distribution repair across four frozen instruction models. On Gemma 2, PCRT reduces the target-kernel error left by AcT and LinEAS on common histories and lowers target-sequence KL on generated histories. In the same sequence protocol, it outperforms a different-content residual control after matching prompt length, response position, candidate entropy, and output-head residual norm. A preregistered AcT confirmation reproduces the sequence result under a second target attribute. On Qwen 2.5, Qwen 3, and Llama 3.1, the same estimator reduces 32-token AcT sequence KL for every test content. Direct source correction is also positive on all three backbones, and the Qwen 2.5 IFEval study connects lower macro target-distribution error to higher macro strict instruction compliance.

The output-head analysis applies to each model through its native vocabulary matrix and output transformation. Gemma 2 uses its bounded softcap, while the three external models use their native linear output interfaces. Sequence evaluation scores each model's low-precision kernel directly. The results support target-distribution matching as an activation-control objective across the tested output interfaces and hidden-state coordinate systems.

### O.2 Method limitations

The current controller is linear in a training-derived residual coordinate. Equation 11 defines the unrestricted measurable optimum within the fixed learned residual span and chosen information set, but the implementation estimates only a shrinkage linear member of that class. A nonlinear conditional mean could help when residuals vary nonlinearly with the candidate state. Such an extension must be chosen on training data and evaluated on fresh test contents.

The repair acts at the final normalized state. This location prevents later Transformer blocks from attenuating the current correction and gives direct access to the output head. It also means that the correction does not rewrite internal keys and values. Tasks that require durable changes to internal computation may benefit from earlier interventions, but they require an explicit propagation analysis.

Same-content target pairs are a real supervision requirement. They can be produced from controlled prompts, paired examples, or task demonstrations, but they may be expensive or ambiguous

for open-ended attributes. The target instruction also defines the desired behavior. A poor target distribution cannot be repaired by a better estimator.

### O.3 Relation to nearby steering goals

Flow-based, feedback, contextual linear, prompt-replacement, language-conditioned flow, and model-based control methods provide expressive online transformations (Jin et al., 2026; Nguyen et al., 2026; Li et al., 2026; Hsu et al., 2026; Heyman & Vandeputte, 2026; Shi et al., 2026; Skifstad et al., 2026). PCRT predicts the residual that remains after a candidate state is produced and can be attached when an editor exposes candidate states and paired targets. Post-editor evidence covers AcT and LinEAS on Gemma 2 and AcT on three additional backbones. Direct source correction is evaluated across those three backbones and on five IFEval categories.

### O.4 Further evaluation

The current evaluation varies one major axis at a time. The Gemma 2 study provides two-editor evidence, the shared COCO study evaluates AcT across three additional backbones, and the public IFEval study evaluates direct correction on Qwen 2.5. The measured distributional horizons are 16 or 32 tokens, and strict behavior is scored on generations of at most 128 tokens. Robustness to lexical or structural paraphrases of the target instruction is also unmeasured. A valid study should define paraphrase families before fitting and evaluate them on untouched contents without selecting prompts on the test set. A complete model-by-task matrix, additional editor classes, and longer sequence horizons remain for future evaluation.

The matched-data comparison separates effective correction from universal controller ranking. The evidence establishes a target-distribution objective, a post-editor correction, and a direct source-correction form with low online cost. Future studies can cross each tested backbone with public instruction tasks, add editors that intervene earlier in the network, and measure longer autoregressive horizons while retaining the same source-target pairing, native output-head geometry, independent-item inference, and official task evaluators.

### O.5 Ethical considerations

Activation steering can alter model behavior without changing stored weights, which can support personalization and controlled deployment. The same property can conceal behavioral changes from users or downstream auditors. A fitted correction should therefore be documented together with its target instruction, training pairs, intervention location, and evaluation scope. Our experiment uses public caption and instruction-following data and does not involve human subjects, personal data collection, or deployment decisions.