COUNTERSTEER: ACTIVATION STEERING FROM UNDESIRABLE EXAMPLES ALONE

Anonymous authors
Paper under double-blind review

ABSTRACT

Activation steering enables inference-time control of large language models by intervening directly on their hidden representations. Existing methods typically derive steering directions by contrasting desired and undesired behavior, which requires a carefully constructed desired dataset that can be difficult or costly to obtain. We introduce COUNTERSTEER, a one-sided activation-steering framework that constructs steering directions using only examples of the behavior to suppress, without requiring desired counterparts or contrastive pairs. Our primary method, COUNTERSTEER-GRAD, constructs a reusable steering vector by reversing the local activation-space direction that would increase the likelihood of undesired examples. It aggregates per-sequence activation gradients of the undesired-data loss in the frozen base model, requiring neither adapter training nor parameter updates. We also introduce COUNTERSTEER-LORA, a variant that goes beyond this one-step probe by training a disposable low-rank adapter on undesired examples and reversing the mean same-input activation shift between the adapted and base models. The adapter is discarded before inference. Both methods apply a precomputed vector to the base model's activations. Theoretically, we establish local suppression for the fixed-weight construction score of COUNTERSTEER-GRAD and characterize the additional alignment condition required for the LoRA-based variant. Across a range of models and behaviors, COUNTERSTEER-GRAD and COUNTERSTEER-LORA match or outperform contrastive steering baselines despite requiring only undesired examples.

1 INTRODUCTION

Activation steering controls the behavior of large language models (LLMs) by modifying their hidden representations during inference (Turner et al., 2023; Rimsky et al., 2024). A common approach constructs a steering vector from the difference in mean activations between desired and undesired examples. More flexible approaches learn transformations between source and target activation distributions (Rodriguez et al., 2025; Abdullaev et al., 2026). These constructions differ in expressiveness, but share a supervision requirement: examples representing both sides of the intended behavioral change. This raises a practical question: *how can we construct a steering direction when examples of the behavior to suppress are available, but suitable desired counterparts are not?*

This asymmetry arises naturally when an undesirable response is reusable, but an appropriate response depends on the request. For illustration, consider two benign questions: "How do I kill a frozen Python process?" and "How do I shoot a portrait indoors?" The same generic response–"I'm sorry, but I can't help with that"–would constitute an inappropriate refusal in either case. Providing helpful alternatives, however, requires different, task-specific answers about computing and photography. More generally, identifying a refusal or a sycophantic response does not necessarily provide the correct response to replace it (Sharma et al., 2024). Constructing desired counterparts may require domain knowledge, factual verification (Min et al., 2023), and prompt-specific annotation. Moreover, differences in content, length, style, and confidence between behavioral collections can enter a contrastive steering direction alongside the intended behavior (Tan et al., 2024). The challenge is therefore not simply to collect more data, but to obtain useful control from the behavioral supervision that is available.

Contributions. We study activation steering under this asymmetric behavioral supervision setting. Specifically, we investigate *one-sided direction construction*, in which a reusable steering vector

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

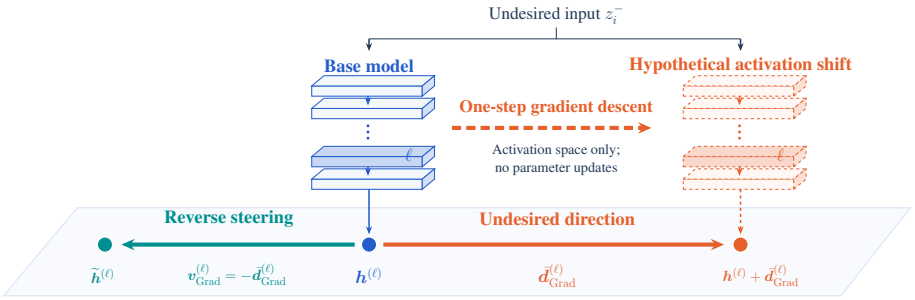


Figure 1: **COUNTERSTEER-GRAD from undesired examples alone.** A conceptual gradient-descent step on the fixed-weight undesired-token NLL defines an activation-space direction $\bar{\boldsymbol{d}}^{(\ell)}_{\text{Grad}}$ that locally favors undesired tokens. Reversing it yields $\boldsymbol{v}^{(\ell)}_{\text{Grad}} = -\bar{\boldsymbol{d}}^{(\ell)}_{\text{Grad}}$, which is added to the base activations with strength α at inference. The right branch illustrates the hypothetical activation step; vector construction requires only gradients, with no parameter updates.

is derived from a collection of undesired examples, without desired counterparts or a contrasting behavioral collection. Our goal is to determine whether examples that specify what the model should avoid can provide an effective direction for inference-time control.

We introduce COUNTERSTEER, a one-sided optimization framework that constructs steering vectors by reversing feature directions associated with undesired behavior. Our primary method, COUNTERSTEER-GRAD, directly extracts a reusable steering vector from activation gradients of the undesired-data loss, without updating model parameters. Specifically, it reverses the local loss-descent direction that increases the likelihood of undesired examples and applies the resulting vector to the frozen base model at inference (Figure 1). We further introduce COUNTERSTEER-LORA, an adaptation-based variant that reverses the mean same-input activation shift induced by a low-rank adapter (LoRA) copy of the base model trained on undesired examples.

Our theoretical analysis characterizes when these interventions locally suppress undesired-token likelihood. For COUNTERSTEER-GRAD, we show that the constructed direction locally decreases its corresponding weighted log-likelihood score when the aggregate gradient is nonzero and the construction and evaluation settings are aligned. For COUNTERSTEER-LORA, we relate the measured representation shift to the adapter-parameter displacement and derive a sufficient alignment condition for effective reversal.

Our contributions are threefold:

1. We introduce COUNTERSTEER, a one-sided activation-steering framework that constructs steering vectors from undesired examples alone. The key idea is to estimate an activation-space direction associated with the undesired behavior and steer the model in the opposite direction.
2. We develop two instantiations of COUNTERSTEER: COUNTERSTEER-GRAD, a training-free method that constructs steering vectors directly from activation gradients, and COUNTERSTEER-LORA, which adapts the model on undesired examples and reverses the resulting representation shift.
3. We provide theoretical guarantees for our framework: for COUNTERSTEER-GRAD, we establish local suppression of its fixed-weight construction score; for COUNTERSTEER-LORA, we characterize adaptation-induced representation shifts and sufficient alignment conditions for local suppression.

We empirically analyze the steering directions constructed by COUNTERSTEER-GRAD and COUNTERSTEER-LORA, showing that both exhibit positive alignment with contrastive steering directions. We further evaluate COUNTERSTEER on over-refusal and sycophancy benchmarks across Qwen, Llama, and Gemma models using additive, angular, and ablation-based interventions. Our results show that steering vectors constructed from undesired examples alone can effectively suppress targeted behaviors, while exposing trade-offs with harmful behavior and general capability.

Organization. Section 2 reviews activation steering and introduces COUNTERSTEER and its two variants. Section 3 provides theoretical analysis. Section 4 presents empirical evaluations, and Sec-

tion 5 provides further analyses and ablations. Section 6 concludes with limitations. The appendix provides algorithmic details, additional analyses, theoretical proofs, and related work (Appendix A).

2 COUNTERSTEER: ACTIVATION STEERING FROM UNDESIRED EXAMPLES

We introduce COUNTERSTEER, a one-sided optimization framework that constructs steering vectors from undesired examples alone. The framework uses an undesired-data fitting objective to identify a candidate feature direction, then reverses it at inference. Our primary method, COUNTERSTEER-GRAD, probes this objective directly in activation space; COUNTERSTEER-LORA instead fits an adapter and measures the resulting activation shift.

2.1 BACKGROUND: ACTIVATION STEERING

Activation steering modifies intermediate representations at inference to control model behavior (Turner et al., 2023; Rimsky et al., 2024). For a hidden state $\boldsymbol{h}^{(\ell)} \in \mathbb{R}^d$ at layer ℓ, an additive intervention is

$$\widetilde{\boldsymbol{h}}^{(\ell)} = \boldsymbol{h}^{(\ell)} + \alpha \boldsymbol{v}^{(\ell)}, \tag{1}$$

where $\boldsymbol{v}^{(\ell)}$ is the steering vector and α controls its strength.

A standard construction uses the difference in mean activations between desired and undesired sets, $\mathcal{D}^+$ and $\mathcal{D}^-$:

$$\boldsymbol{v}^{(\ell)}_{\mathrm{DiM}} = \mathbb{E}_{z\sim\mathcal{D}^+}[\boldsymbol{h}^{(\ell)}_{\boldsymbol{\theta}_0}(z)] - \mathbb{E}_{z\sim\mathcal{D}^-}[\boldsymbol{h}^{(\ell)}_{\boldsymbol{\theta}_0}(z)], \tag{2}$$

where $\boldsymbol{\theta}_0$ denotes the fixed base model and the expectations are empirical averages under a common extraction rule. Contrastive Activation Addition (CAA) estimates such directions from paired examples (Rimsky et al., 2024), although the formulation also applies to unpaired sets. Distribution-based methods learn richer steering maps, but similarly rely on contrasting source and target activation distributions (Rodriguez et al., 2025; Abdullaev et al., 2026; Li et al., 2026).

2.2 THE COUNTERSTEER FRAMEWORK

A shared undesired-behavior objective. Let $p_{\boldsymbol{\theta}_0}$ be the base language model and $\mathcal{D}^- = \{z_i^-\}_{i=1}^{N_-}$ a nonempty collection of undesired token sequences. For each sequence, let $\mathcal{T}_i$ be a nonempty set of valid next-token prediction indices selected for scoring. For prompt–response pairs $z_i^- = [x_i; y_i^-]$, $\mathcal{T}_i$ contains only response-token indices; for standalone text, it contains all valid prediction indices or a designated target span. Padding tokens are excluded.

For a predictive distribution p, define the masked mean token negative log-likelihood (NLL)

$$\mathcal{L}_i(p) = -\frac{1}{|\mathcal{T}_i|} \sum_{t\in\mathcal{T}_i} \log p\big(z^-_{i,t} \mid z^-_{i,<t}\big). \tag{3}$$

Here $z^-_{i,t}$ denotes the token at position t and $z^-_{i,<t}$ its preceding context. Tokens and scoring masks remain fixed under teacher forcing. Reducing $\mathcal{L}_i$ increases the mean log-likelihood of the scored undesired tokens. Unscored context tokens incur no direct prediction loss but may affect subsequent scored tokens.

Auxiliary fitting and reversal. Let $\boldsymbol{\xi} \in \Xi$ parameterize a temporary modification of the base model, with reference state $\boldsymbol{\xi}_0$ satisfying $p_{\boldsymbol{\xi}_0} = p_{\boldsymbol{\theta}_0}$. Both methods derive from the auxiliary fitting problem

$$\min_{\boldsymbol{\xi}\in\Xi} \mathcal{L}_-(p_{\boldsymbol{\xi}}), \qquad \mathcal{L}_-(p) := \frac{1}{N_-}\sum_{i=1}^{N_-} \mathcal{L}_i(p). \tag{4}$$

Fitting is a construction device, not the deployed intervention. We use either a local descent direction or a finite optimization trajectory, without requiring convergence to a minimizer. Let $\bar{\boldsymbol{d}}^{(\ell)}$ denote the feature direction or mean displacement extracted from this probe at layer ℓ. COUNTERSTEER reverses it by setting $\boldsymbol{v}^{(\ell)} = -\bar{\boldsymbol{d}}^{(\ell)}$. The choice of optimization variables and extraction rule yields our two methods: COUNTERSTEER-GRAD uses a shared activation offset, whereas COUNTERSTEER-LORA uses adapter parameters.

Shared inference-time intervention. After construction, we apply the reverse vector to the original base model, without retaining the temporary modification. For the decoder-block output $\boldsymbol{h}_t^{(\ell)}$ at position t, the intervention follows Eq. 1: $\widetilde{\boldsymbol{h}}_t^{(\ell)} = \boldsymbol{h}_t^{(\ell)} + \alpha\boldsymbol{v}^{(\ell)}$, $\alpha \geq 0$. We apply the same displacement to all prompt tokens during prefill and each newly processed token during decoding. Setting $\alpha = 0$ recovers the base model. Neither variant requires an active adapter or gradient computation during generation.

2.3 COUNTERSTEER-GRAD: REVERSING ACTIVATION GRADIENTS

Our primary method, COUNTERSTEER-GRAD, instantiates Eq. 4 directly in activation space. Let $\boldsymbol{u} \in \mathbb{R}^d$ be a shared additive offset at layer ℓ, and let $p^{(\ell)}_{\boldsymbol{\theta}_0,\boldsymbol{u}}$ denote the modified model. All model parameters remain frozen, and $\boldsymbol{u} = \mathbf{0}$ recovers the base model. Thus, the auxiliary optimization variable is $\boldsymbol{\xi} = \boldsymbol{u}$.

For sequence z_i^-, let $\mathcal{K}_i$ contain the non-padding hidden-state positions receiving the offset. The sets $\mathcal{T}_i$ and $\mathcal{K}_i$ select scored tokens and intervention sites, respectively. Writing $\mathcal{L}_i = \mathcal{L}_i(p_{\boldsymbol{\theta}_0})$, the chain rule gives

$$\boldsymbol{q}_i^{(\ell)} := \nabla_{\boldsymbol{u}} \mathcal{L}_i\big(p^{(\ell)}_{\boldsymbol{\theta}_0,\boldsymbol{u}}\big)\Big|_{\boldsymbol{u}=\mathbf{0}} = \sum_{k\in\mathcal{K}_i} \nabla_{\boldsymbol{h}^{(\ell)}_{i,k}} \mathcal{L}_i. \tag{5}$$

The sum arises because the same offset is applied at every intervention site. These gradients include downstream effects, so unscored context positions may contribute. We evaluate them under teacher forcing with dropout disabled.

A hypothetical gradient-descent step on the shared fitting objective, starting at $\boldsymbol{u} = \mathbf{0}$, is

$$\boldsymbol{u}_{\text{fit}} = -\eta\, \nabla_{\boldsymbol{u}} \mathcal{L}_-\big(p^{(\ell)}_{\boldsymbol{\theta}_0,\boldsymbol{u}}\big)\Big|_{\boldsymbol{u}=\mathbf{0}} = -\frac{\eta}{N_-}\sum_{i=1}^{N_-} \boldsymbol{q}_i^{(\ell)}, \qquad \eta > 0. \tag{6}$$

Rather than apply this fitting step, COUNTERSTEER-GRAD reverses its direction. Absorbing η into the inference-time strength gives

$$\bar{\boldsymbol{d}}^{(\ell)}_{\text{Grad}} = \frac{\boldsymbol{u}_{\text{fit}}}{\eta} = -\frac{1}{N_-}\sum_{i=1}^{N_-} \boldsymbol{q}_i^{(\ell)}, \quad \boldsymbol{v}^{(\ell)}_{\text{Grad}} = -\bar{\boldsymbol{d}}^{(\ell)}_{\text{Grad}} = \frac{1}{N_-}\sum_{i=1}^{N_-} \boldsymbol{q}_i^{(\ell)}. \tag{7}$$

The unnormalized vector is therefore the gradient of the shared NLL with respect to the activation offset. When nonzero, it is a local ascent direction for this loss, equivalently a descent direction for the corresponding mean log-likelihood. Construction requires backpropagation but no parameter updates or adapter fitting.

Per-sequence normalization. In practice, we normalize each sequence gradient before averaging to limit the influence of large-norm examples. With the gradients from Eq. (5), define

$$w_i^{(\ell)} = \frac{1}{\|\boldsymbol{q}_i^{(\ell)}\|_2 + \epsilon}, \qquad \boldsymbol{v}^{(\ell)}_{\text{Grad}} = \frac{1}{N_-}\sum_{i=1}^{N_-} w_i^{(\ell)} \boldsymbol{q}_i^{(\ell)}, \qquad \epsilon > 0. \tag{8}$$

Here normalization is applied after summing over intervention sites, not to individual token gradients. The weights are evaluated once at the unsteered base model and then held fixed. This construction has the same reversal interpretation as the unnormalized derivation: using the per-example NLL in Eq. (3), define

$$\mathcal{L}^{(\ell)}_{-,w}(\boldsymbol{u}) := \frac{1}{N_-}\sum_{i=1}^{N_-} w_i^{(\ell)} \mathcal{L}_i\big(p^{(\ell)}_{\boldsymbol{\theta}_0,\boldsymbol{u}}\big), \qquad \boldsymbol{v}^{(\ell)}_{\text{Grad}} = \nabla_{\boldsymbol{u}} \mathcal{L}^{(\ell)}_{-,w}(\boldsymbol{u})\Big|_{\boldsymbol{u}=\mathbf{0}}. \tag{9}$$

Thus, the normalized implementation reverses the local descent direction $\bar{\boldsymbol{d}}^{(\ell)}_{\text{Grad}} = -\boldsymbol{v}^{(\ell)}_{\text{Grad}}$ of an example-weighted version of the shared NLL; no derivative passes through the weights. We deploy $\boldsymbol{v}^{(\ell)}_{\text{Grad}}$ through Eq. (1). Per-sequence normalization does not make the aggregate a unit vector; any final rescaling is specified separately. Figure 1 and Algorithm 1 summarize the procedure, and Section 3.1 analyzes this normalized construction.

2.4 COUNTERSTEER-LORA: REVERSING ADAPTATION-INDUCED SHIFTS

Moving beyond one-step gradient updates, our second method, COUNTERSTEER-LORA, instantiates the same fitting problem in adapter-parameter space. Let $p_{\boldsymbol{\theta}_0,\boldsymbol{\phi}}$ denote the base model with LoRA parameters $\boldsymbol{\phi}$ (Hu et al., 2021), initialized at $\boldsymbol{\phi}_0$ to recover the base model. Setting $\boldsymbol{\xi} = \boldsymbol{\phi}$ in Eq. 4 gives

$$\mathcal{L}_-(\boldsymbol{\phi}) := \mathcal{L}_-(p_{\boldsymbol{\theta}_0,\boldsymbol{\phi}}) = \frac{1}{N_-}\sum_{i=1}^{N_-} \mathcal{L}_i(p_{\boldsymbol{\theta}_0,\boldsymbol{\phi}}). \tag{10}$$

We optimize this loss for T steps to obtain ϕ_T, keeping the base parameters θ_0 fixed. Unlike COUNTERSTEER-GRAD, which reads a local activation-space direction, COUNTERSTEER-LORA performs finite adaptation and measures its effect on representations.

For extraction inputs $\mathcal{X}_{\text{extract}} = \{\bar{x}_j\}_{j=1}^{M}$ constructed from the available data, let $\boldsymbol{h}_{\phi}^{(\ell)}(x)$ denote the decoder-block output at the final non-padding token of input x. Using identical inputs, tokenization, and readout positions in evaluation mode, we measure

$$\boldsymbol{d}^{(\ell)}(x) = \boldsymbol{h}_{\phi_T}^{(\ell)}(x) - \boldsymbol{h}_{\phi_0}^{(\ell)}(x). \tag{11}$$

In our prompt–response experiments, extraction uses formatted prompts without their response targets, as detailed in Section 4. Averaging and reversing the shifts gives

$$\bar{\boldsymbol{d}}^{(\ell)} = \frac{1}{M}\sum_{j=1}^{M} \boldsymbol{d}^{(\ell)}(\bar{x}_j), \qquad \boldsymbol{v}^{(\ell)} = -\bar{\boldsymbol{d}}^{(\ell)}. \tag{12}$$

We discard the adapter and apply this vector through Eq. 1. Figure 5 illustrates the construction, and Appendix C provides pseudocode.

COUNTERSTEER-GRAD vs. COUNTERSTEER-LORA. Both share the NLL-based reversal principle but yield different directions. For COUNTERSTEER-LORA, reduced adapter loss alone does not guarantee effective reversal; Section 3.2 derives the required alignment condition.

3 THEORETICAL ANALYSIS

Section 2.2 uses a masked-NLL fitting objective to identify a direction and reverses it at inference. We analyze the per-sequence-normalized COUNTERSTEER-GRAD construction in Section 2.3, then the adaptation-induced shift in Section 2.4. Both analyses use the shared activation offset and additive intervention defined in Section 2.

Fix a layer ℓ and a vector $\boldsymbol{v}^{(\ell)} = -\bar{\boldsymbol{d}}^{(\ell)}$, and write $p_\alpha = p^{(\ell)}_{\theta_0,\alpha\boldsymbol{v}^{(\ell)}}$ for the base model under Eq. (1). For a finite, nonempty evaluation collection $\mathcal{E}^-$, let $\mathcal{L}_z$ be the masked NLL in Eq. (3) for sequence z. Define

$$F(\alpha) = -\frac{1}{|\mathcal{E}^-|}\sum_{z\in\mathcal{E}^-} \mathcal{L}_z(p_\alpha), \qquad S_z(\alpha) := -\mathcal{L}_z(p_\alpha). \tag{13}$$

Thus, F is the mean log-likelihood of the scored undesired tokens. On the construction data with matching masks and intervention sites, $F(\alpha) = -\mathcal{L}_-(p_\alpha)$ from Eq. (4). All tokens, scoring masks, intervention sites, weights, and vectors remain fixed as α varies under teacher forcing. The results concern additive steering on fixed sequences, not sampled behavior or capability retention.

A common condition for reversal. Assuming differentiability with respect to the shared offset, Eq. (5) gives

$$\boldsymbol{b}^{(\ell)} := -\frac{1}{|\mathcal{E}^-|}\sum_{z\in\mathcal{E}^-} \nabla_{\boldsymbol{u}}\mathcal{L}_z\big(p^{(\ell)}_{\theta_0,\boldsymbol{u}}\big)\Big|_{\boldsymbol{u}=\boldsymbol{0}}, \qquad F'(0) = \langle \boldsymbol{b}^{(\ell)}, \boldsymbol{v}^{(\ell)}\rangle. \tag{14}$$

This derivative includes every intervention site and its downstream contributions to the score.

Proposition 1 (Local suppression)**.** *Suppose F is twice continuously differentiable on $[0, a]$, with $a > 0$ and $|F''(\alpha)| \le K_F$ for some $K_F > 0$. Define $m^{(\ell)} = -\langle \boldsymbol{b}^{(\ell)}, \boldsymbol{v}^{(\ell)}\rangle = \langle \boldsymbol{b}^{(\ell)}, \bar{\boldsymbol{d}}^{(\ell)}\rangle$. Then*

$$F(\alpha) - F(0) \le -\alpha m^{(\ell)} + \frac{K_F}{2}\alpha^2, \qquad 0 \le \alpha \le a. \tag{15}$$

If $m^{(\ell)} > 0$, every $0 < \alpha < \min\{a, 2m^{(\ell)}/K_F\}$ strictly decreases F.

The guarantee is local because the first-order decrease must dominate curvature over the stated range of strengths. Appendix F.1 gives the proof.

3.1 LOCAL SUPPRESSION WITH NORMALIZED COUNTERSTEER-GRAD

The final construction in Section 2.3 is the gradient of the fixed-weight NLL in Eq. (9). Its matched score is therefore the negative of that same objective:

$$F_w(\alpha) := -\mathcal{L}^{(\ell)}_{-,w}\big(\alpha \boldsymbol{v}^{(\ell)}_{\text{Grad}}\big) = -\frac{1}{N_-}\sum_{i=1}^{N_-} w_i^{(\ell)}\mathcal{L}_i(p_\alpha), \tag{16}$$

where p_α now uses the normalized construction vector $\boldsymbol{v}^{(\ell)}_{\mathrm{Grad}}$. The weights arise solely from per-sequence normalization; the scored tokens and per-example NLL are unchanged from Section 2.2.

Corollary 1 (Local suppression by normalized COUNTERSTEER-GRAD)**.** *Use the construction examples, scoring masks, and intervention sites from Section 2.3, and hold $w_i^{(\ell)}$ fixed at their base-model values. For the vector in Eq. (8),*

$$F'_w(0) = -\|\boldsymbol{v}^{(\ell)}_{\mathrm{Grad}}\|_2^2. \tag{17}$$

If F_w is twice continuously differentiable on $[0, a_w]$, with $a_w > 0$ and $|F''_w(\alpha)| \le K_w$ for some $K_w > 0$, then

$$F_w(\alpha) - F_w(0) \le -\alpha\|\boldsymbol{v}^{(\ell)}_{\mathrm{Grad}}\|_2^2 + \frac{K_w}{2}\alpha^2, \qquad 0 \le \alpha \le a_w. \tag{18}$$

A nonzero vector strictly decreases F_w whenever $0 < \alpha < \min\{a_w, 2\|\boldsymbol{v}^{(\ell)}_{\mathrm{Grad}}\|_2^2/K_w\}$. A zero vector produces no intervention.

The derivative identity follows directly from $\boldsymbol{v}^{(\ell)}_{\mathrm{Grad}} = \nabla_{\boldsymbol{u}}\mathcal{L}^{(\ell)}_{-,w}(\mathbf{0})$; Proposition 1 then gives the bound. The guarantee concerns the implemented per-sequence normalization and its score F_w; it does not imply a decrease in the unweighted F, whose derivative can have either sign. For a nonzero aggregate, final unit-vector rescaling preserves the negative derivative, changing its value to $-\|\boldsymbol{v}^{(\ell)}_{\mathrm{Grad}}\|_2$ for the rescaled score. Appendix F.2 provides the proof and rescaled bound.

3.2 ALIGNMENT OF ADAPTATION-INDUCED SHIFTS

Section 2.4 fits the unweighted NLL in Section 2.2 through adapter parameters. Let $\Delta\boldsymbol{\phi} = \boldsymbol{\phi}_T - \boldsymbol{\phi}_0$. Suppose each representation map $\boldsymbol{h}^{(\ell)}_{\boldsymbol{\phi}}(\bar{x}_j)$ is continuously differentiable near the segment joining these adapter states, with an $L^{(\ell)}_j$-Lipschitz parameter Jacobian on that segment, in the operator norm relative to the Euclidean parameter norm. Let $\bar{\boldsymbol{J}}^{(\ell)}$ be the mean of these Jacobians at $\boldsymbol{\phi}_0$ and $\bar{L}^{(\ell)} = M^{-1}\sum_j L^{(\ell)}_j$. Averaging the Taylor expansions of Eq. (11) yields

$$\bar{\boldsymbol{d}}^{(\ell)} = \bar{\boldsymbol{J}}^{(\ell)}\Delta\boldsymbol{\phi} + \bar{\boldsymbol{r}}^{(\ell)}, \qquad \|\bar{\boldsymbol{r}}^{(\ell)}\|_2 \le \frac{\bar{L}^{(\ell)}}{2}\|\Delta\boldsymbol{\phi}\|_2^2, \tag{19}$$

where $\bar{\boldsymbol{r}}^{(\ell)}$ is the mean Taylor remainder. For the reverse vector in Eq. (12),

$$m^{(\ell)} \ge \langle \boldsymbol{b}^{(\ell)}, \bar{\boldsymbol{J}}^{(\ell)}\Delta\boldsymbol{\phi}\rangle - \frac{\bar{L}^{(\ell)}}{2}\|\boldsymbol{b}^{(\ell)}\|_2\|\Delta\boldsymbol{\phi}\|_2^2. \tag{20}$$

A positive right-hand side suffices for local suppression under Proposition 1. Normalized COUNTERSTEER-GRAD is aligned with its matched weighted objective by construction, whereas COUNTERSTEER-LORA requires additional alignment with the chosen score. Reducing adapter loss alone does not establish this alignment: adapter fitting changes parameters, whereas deployment uses an averaged single-layer activation shift. Derivations and extraction-to-intervention transfer bounds are given in Appendices F.3 and F.4.

4 EXPERIMENTS

CounterSteer constructs steering vectors from undesired examples alone, without desired counterparts or a contrasting behavioral collection. We focus on contrastive baselines to assess whether undesired-only construction can provide effective alternatives to directions derived from two behavioral collections.

4.1 OVER-REFUSAL SUPPRESSION

Over-refusal is a natural testbed: undesired refusals are readily identifiable, while helpful alternatives require prompt-specific construction. Following prior work on steering-based mitigation (Cao et al., 2025) and broader evaluation (Goyal & Daumé III, 2026), we assess whether COUNTERSTEER reduces benign refusal while monitoring harmful responses and general capability.

Setup. We construct COUNTERSTEER vectors from benign prompts paired with undesired refusal responses, without helpful counterparts. We compare both variants with the unsteered model, difference-in-means (DiM) (see Section 2.1), and LAT-PCA (Zou et al., 2023a) on Qwen2.5-3B-Instruct, Qwen2.5-7B-Instruct (Qwen et al., 2025), Llama-3.1-8B-Instruct (Grattafiori et al., 2024),

Table 1: **Over-refusal steering results on held-out test sets.** Macro and micro averages aggregate the four behavioral rates with equal and sample-count weights, respectively. Unless otherwise noted, **bold**/underlining mark best/second-best results, including ties; blue shading marks our methods.

Model	Method	Benign Refusal ↓		Harmful Behavior ↓		Average Rates ↓		Capability
		OR-Hard (%)	XSTest (%)	HarmBench (%)	OR-Toxic (%)	Macro avg (%) ↓	Micro avg (%) ↓	tinyBench (%) ↑
Qwen2.5-3B Instruct	Original (no steer)	48.3	15.2	**19.0**	**1.00**	20.9	29.9	57.7
	DiM	27.7	10.8	25.3	2.5	16.6	20.0	**58.8**
	LAT-PCA	27.6	10.8	25.0	2.5	16.5	19.9	58.0
	CounterSteer-Grad (Ours)	**23.6**	**4.0**	27.0	2.5	**14.3**	**17.3**	57.3
	CounterSteer-LoRA (Ours)	25.2	5.2	26.7	2.76	15.0	18.3	58.0
Qwen2.5-7B Instruct	Original (no steer)	17.1	4.4	**27.3**	**3.8**	13.2	14.3	**65.2**
	DiM	11.6	3.2	28.7	4.8	12.1	11.8	**65.2**
	LAT-PCA	12.9	**2.8**	31.3	4.8	12.9	12.8	64.3
	CounterSteer-Grad (Ours)	11.5	3.2	30.3	4.5	12.4	11.9	64.7
	CounterSteer-LoRA (Ours)	**9.0**	**2.8**	31.3	4.26	**11.8**	**10.7**	64.3
Llama-3.1-8B Instruct	Original (no steer)	28.1	7.6	**29.7**	**5.51**	17.7	21.1	**62.2**
	DiM	19.3	**5.2**	33.0	8.0	16.4	17.3	60.7
	LAT-PCA	23.9	6.4	32.3	8.52	17.8	19.8	60.7
	CounterSteer-Grad (Ours)	**18.4**	6.0	32.3	8.52	**16.3**	**16.9**	61.3
	CounterSteer-LoRA (Ours)	20.9	6.0	34.3	6.52	16.9	18.1	60.7
Gemma-2-9B it	Original (no steer)	79.2	17.6	**7.0**	**0.75**	26.1	44.1	**65.0**
	DiM	38.3	8.8	15.7	3.0	16.4	23.8	64.3
	LAT-PCA	38.3	8.0	16.0	2.8	16.3	23.7	**65.0**
	CounterSteer-Grad (Ours)	39.2	8.0	14.0	2.0	15.8	23.7	**65.0**
	CounterSteer-LoRA (Ours)	**23.0**	**2.8**	18.7	3.5	**12.0**	**15.8**	64.3

Table 2: **Capability retention on tinyBenchmarks.** Accuracy (%, higher is better) under the steering configurations in Table 1.

Model	Method	tMMLU	tARC-C	tHellaSwag	tWinogrande	tTruthfulQA	tGSM8k	Avg.
Qwen2.5-3B Instruct	Original (no steer)	63.0	58.0	**47.0**	56.0	**42.0**	80.0	57.7
	DiM	**64.0**	57.0	**47.0**	58.0	**42.0**	**85.0**	**58.8**
	LAT-PCA	63.0	**61.0**	**47.0**	**61.0**	37.0	79.0	58.0
	CounterSteer-Grad (Ours)	**64.0**	57.0	46.0	58.0	40.0	79.0	57.3
	CounterSteer-LoRA (Ours)	63.0	55.0	**47.0**	**61.0**	**42.0**	80.0	58.0
Qwen2.5-7B Instruct	Original (no steer)	76.0	64.0	**46.0**	71.0	42.0	**92.0**	**65.2**
	DiM	76.0	**67.0**	44.0	**72.0**	43.0	89.0	**65.2**
	LAT-PCA	**77.0**	64.0	45.0	70.0	41.0	89.0	64.3
	CounterSteer-Grad (Ours)	76.0	66.0	45.0	70.0	**44.0**	87.0	64.7
	CounterSteer-LoRA (Ours)	75.0	63.0	45.0	**72.0**	42.0	89.0	64.3
Llama-3.1-8B Instruct	Original (no steer)	**63.0**	**63.0**	**48.0**	**75.0**	40.0	**84.0**	**62.2**
	DiM	62.0	**63.0**	**48.0**	72.0	39.0	80.0	60.7
	LAT-PCA	62.0	**63.0**	**48.0**	72.0	39.0	80.0	60.7
	CounterSteer-Grad (Ours)	**63.0**	**63.0**	**48.0**	73.0	40.0	81.0	61.3
	CounterSteer-LoRA (Ours)	62.0	**63.0**	**48.0**	71.0	**41.0**	79.0	60.7
Gemma-2-9B it	Original (no steer)	**76.0**	70.0	47.0	66.0	43.0	**88.0**	**65.0**
	DiM	73.0	71.0	46.0	69.0	41.0	86.0	64.3
	LAT-PCA	73.0	71.0	**48.0**	**71.0**	42.0	85.0	**65.0**
	CounterSteer-Grad (Ours)	74.0	70.0	47.0	68.0	**44.0**	87.0	**65.0**
	CounterSteer-LoRA (Ours)	75.0	**75.0**	**48.0**	69.0	37.0	82.0	64.3

and Gemma-2-9B-it (Team et al., 2024). We evaluate benign refusal on OR-Hard (Cui et al., 2025) and XSTest (Röttger et al., 2024) using WildGuard (Han et al., 2024), harmful responses on HarmBench (Mazeika et al., 2024) and OR-Toxic (Cui et al., 2025), and capability on tinyBenchmarks (Maia Polo et al., 2024). Layers and strengths are selected on labeled validation data.

Results. Both variants reduce benign refusal across all four models (Table 1). CounterSteer-Grad achieves the lowest OR-Hard refusal on Qwen2.5-3B and Llama-3.1-8B, while CounterSteer-LoRA leads on both benign-refusal benchmarks on Qwen2.5-7B and Gemma-2-9B. Lower benign refusal accompanies increased harmful responses. Aggregate rates should therefore be read alongside the separate benign and harmful outcomes. Average tinyBenchmarks accuracy stays within 1 and 1.5 points of the base model for CounterSteer-Grad and CounterSteer-LoRA, respectively (Table 2).

4.2 Sycophancy

Setup. We construct CounterSteer vectors from 512 sycophantic responses per model, covering answer agreement, pushback, and feedback (Joshi et al., 2017; Sharma et al., 2024). On Qwen2.5-

Table 3: **Sycophancy steering results on held-out test sets.** Behavioral metrics follow Sharma et al. (2024); capability is measured on tinyBenchmarks. Random is excluded from ranking.

Model	Method	Answer Sycophancy		Feedback Sycophancy	Are You Sure?	Capability
		Syc. Agreement (%) ↓	Correction (%) ↑	Positivity Gap (pp) ↓	Flip (%) ↓	tinyBench (%) ↑
Qwen2.5-3B Instruct	Original (no steer)	34.7	46.5	78.1	37.5	57.7
	DiM	23.5	50.8	83.6	42.2	59.3
	LAT-PCA	25.9	49.1	81.6	39.8	**59.5**
	Random	31.1	47.4	81.0	59.1	58.5
	CounterSteer-Grad (Ours)	23.1	**52.8**	73.7	**4.4**	58.0
	CounterSteer-LoRA (Ours)	**22.0**	49.2	**64.9**	11.9	58.7
Qwen2.5-7B Instruct	Original (no steer)	27.8	57.3	62.2	18.5	**65.2**
	DiM	35.3	53.0	64.3	**10.0**	61.0
	LAT-PCA	39.7	49.7	61.6	10.5	61.3
	Random	25.7	56.5	62.0	18.3	63.5
	CounterSteer-Grad (Ours)	**16.2**	**64.2**	**52.9**	10.5	64.3
	CounterSteer-LoRA (Ours)	20.9	59.9	60.9	11.1	64.7
Llama-3.1-8B Instruct	Original (no steer)	15.0	72.4	85.2	75.3	**62.2**
	DiM	15.4	71.8	82.7	74.7	61.5
	LAT-PCA	14.9	72.0	85.1	73.2	61.0
	Random	15.1	71.8	85.6	75.0	60.7
	CounterSteer-Grad (Ours)	15.3	71.5	82.8	**66.3**	60.7
	CounterSteer-LoRA (Ours)	**13.9**	**72.7**	**81.7**	72.3	61.7

Table 4: **Steering-vector comparison across intervention types.** Refusal and Harmful denote refusal on OR-Hard and harmful responses on HarmBench (%), respectively; lower is better. Only best results are bolded.

Method	Addition (ActAdd)		Rotation (Angular Steering)		Ablation (DirAbl)	
	Refusal ↓	Harmful ↓	Refusal ↓	Harmful ↓	Refusal ↓	Harmful ↓
DiM	27.7	**25.3**	33.1	27.3	**0.0**	81.7
CounterSteer-Grad (Ours)	**23.6**	27.0	**27.1**	31.0	9.1	**37.3**
CounterSteer-LoRA (Ours)	25.2	26.7	28.3	**27.0**	0.1	64.0

3B/7B-Instruct and Llama-3.1-8B-Instruct (Qwen et al., 2025; Grattafiori et al., 2024), we compare with the base model, DiM, LAT-PCA, and random steering. Held-out tests measure agreement, correction, positivity gaps, and answer flips using validation-selected layers and strengths.

Results. Both variants improve three behaviors and answer recovery on Qwen with one configuration each (Table 3); DiM and LAT-PCA show mixed effects. CounterSteer-Grad leads overall on Qwen2.5-7B; on Qwen2.5-3B and better reduces flips, while CounterSteer-LoRA better reduces agreement and feedback bias. Llama gains are smaller: CounterSteer-LoRA improves all three behaviors; CounterSteer-Grad reduces flips but slightly worsens answer-level results.

4.3 Integrating CounterSteer with different intervention types

Setup. We compare CounterSteer-LoRA, CounterSteer-Grad, and DiM under activation addition, angular steering (Vu & Nguyen, 2025), and directional ablation (Arditi et al., 2024), evaluating benign refusal and harmful-response rates.

Results. Table 4 shows that both CounterSteer variants reduce benign refusal more than DiM under addition and rotation. Under rotation, CounterSteer-Grad achieves lower refusal than CounterSteer-LoRA but higher harmful-response rates. Under directional ablation, CounterSteer-Grad yields fewer harmful responses than both alternatives but more benign refusals. Thus, both the vector source and intervention type shape the refusal–safety trade-off.

5 Analysis and Ablations

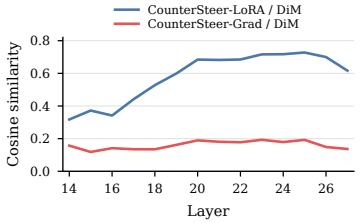


Figure 2: **Alignment with DiM.** Layerwise cosine similarity on Qwen2.5-7B-Instruct.

Steering-direction alignment. We compare CounterSteer with DiM to examine whether steering from undesired examples alone yields directions similar to those from contrastive data. Figure 2 reports layerwise cosine similarity on Qwen2.5-7B-Instruct, with all vectors oriented toward non-refusal. Both variants align positively with DiM, more strongly for CounterSteer-LoRA, while remaining distinct. Details are in Appendix E.

Target ablation. We compare refusal-target reversal with helpful-target steering on identical prompts. Positive-target CounterSteer-Grad negates normalized NLL gradients, while CounterSteer-LoRA uses forward adaptation shifts with the same budget. At matched strength, reversal achieves higher non-refusal across most layers for both variants (Figure 3). Helpful targets also require prompt-specific construction and may capture content or style beyond refusal behavior. Additional results are in Appendix E.1.

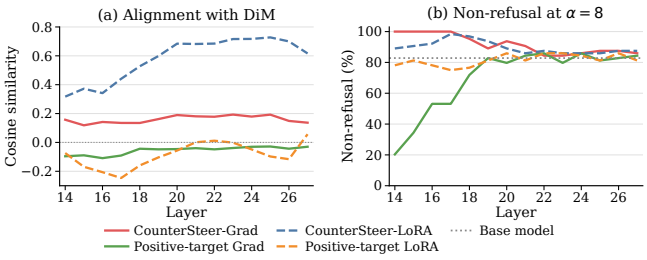


Figure 3: **Target ablation on Qwen2.5-7B-Instruct. (a)** Alignment with DiM. **(b)** Validation non-refusal at $\alpha = 8$. Solid: Grad; dashed: LoRA; dotted: base model.

Refusal–safety trade-off. Figure 4 compares the methods on Qwen2.5-7B-Instruct at a fixed layer over common steering strengths. Stronger steering generally reduces benign refusal while increasing harmful behavior. CounterSteer-Grad achieves greater refusal reduction at higher strengths, whereas CounterSteer-LoRA follows a trade-off closer to DiM. These results highlight steering strength as a control over the balance between refusal reduction and safety. Evaluation details are provided in Appendix E.2.

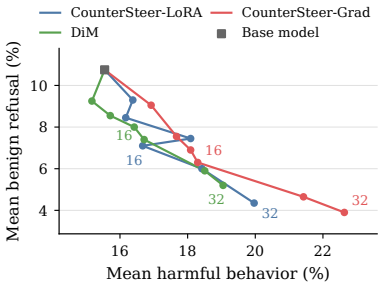


Figure 4: **Refusal–safety trade-off.** Layer-23 sweep; lower left is better.

Inference efficiency. Both variants apply a precomputed vector through the same single-layer additive intervention. Measured mean latency overhead on Qwen2.5-7B-Instruct is at most 1.1% across the tested batch sizes (Appendix E.3).

Steering with Momentum. We test CounterSteer-Mom, which accumulates gradients across activation-offset updates toward undesired responses before reversing this direction with frozen model weights (Appendix E.4). On Qwen2.5-3B-Instruct, it reduces incorrect-answer agreement and improves flips and correction, but slightly increases feedback bias (Table 11 in Appendix E.4). It underperforms CounterSteer-Grad on all metrics and CounterSteer-LoRA on all except correction, showing no benefit over CounterSteer-Grad in this preliminary setting.

Steering a newer, larger model. We evaluate CounterSteer on Qwen3.6-27B (Qwen Team, 2026), a newer, larger dense model combining Gated DeltaNet (Yang et al., 2025b) and full-attention layers. Table 12 in Appendix E.5 shows that both variants substantially reduce benign refusal. LoRA achieves the lowest OR-Hard refusal rate, matches DiM on XSTest, and yields fewer harmful responses than Grad on both safety benchmarks. Grad also reduces OR-Hard refusal slightly more than DiM, but with a higher HarmBench harmful-response rate. All steering methods increase harmful responses over the unsteered model. These results extend the observed refusal-reduction effects to this newer hybrid model, with a refusal–safety trade-off consistent with the main experiments.

6 Conclusion and Limitations

We introduced CounterSteer, a framework for constructing steering vectors from undesired examples alone. Our primary method, CounterSteer-Grad, reverses a local activation-space fitting direction without updating model parameters, while CounterSteer-LoRA reverses adaptation-induced representation shifts. Our analysis establishes local suppression for the matched weighted score and characterizes the alignment required for adaptation-based reversal. Experiments on over-refusal and sycophancy show that CounterSteer can match or outperform contrastive steering baselines using only undesired examples for vector construction.

Limitations and future work. Our current study focuses on one-step activation gradients and adaptation-induced shifts. Preliminary exploration of momentum-based, multistep construction has not shown consistent improvements over CounterSteer-Grad. Future work could investigate alternative update rules and aggregation strategies to better exploit multistep optimization for steering-vector construction. Another direction is to jointly steer multiple layers using their respective activation gradients from a single backward pass, potentially enabling finer control over the model's representation trajectory.

AI USE STATEMENT

In this work, we used generative AI tools for providing feedback on our experiments and implementing methods. We have not used generative AI tools for helping develop theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, proposing or refining hypotheses, designing or providing feedback on research methodology, and interpreting results, and the rest of the required disclosure tasks are not applicable to this work. Additionally, we used generative AI tools for modifying scientific figures, suggesting experimental parameters, editing software code, editing our research paper to improve readability, and identifying relevant literature. We have reviewed all AI-assisted work. In particular, we discussed and evaluated within our group the AI-generated suggestions for experimental design. We also carefully assessed AI suggestions for method implementation and wrote the final code ourselves. In addition, we reviewed all AI-generated or AI-edited figures and manuscript revisions, and carefully read and verified the literature suggested by AI. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

Given the nature of the work, we do not foresee any negative societal and ethical impacts of our work. However, risks remain: By eliminating the need to collect desired examples, COUNTERSTEER may inadvertently lower the barrier to generating nuanced harmful content, such as persuasive misinformation or biased narratives. Although it does not expand the baseline risk profile of LLMs, robust safeguards, transparency, accountability, and ongoing ethical review are required for responsible use.

REPRODUCIBILITY STATEMENT

Source codes for our experiments are provided in the supplementary materials of the paper. The details of our experimental settings are given in Section 4 and the Appendix. All datasets that we used in the paper are published, and they are easy to access in the Internet.

REFERENCES

Laziz U. Abdullaev, Noelle Y. L. Wong, Ryan T. Z. Lee, Shiqi Jiang, Khoi N. M. Nguyen, and Tan M. Nguyen. Concept heterogeneity-aware representation steering. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2603.02237.

Sina Alemohammad, Zhangyang Wang, and Richard G. Baraniuk. Neon: Negative extrapolation from self-training improves image generation. *arXiv preprint arXiv:2510.03597*, 2025. URL https://arxiv.org/abs/2510.03597.

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in Language Models Is Mediated by a Single Direction. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://openreview.net/forum?id=pH3XAQME6c.

Zouying Cao, Yifei Yang, and Hai Zhao. Scans: Mitigating the exaggerated safety for llms via safety-conscious activation steering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 23523–23531, 2025.

Justin Cui, Wei-Lin Chiang, Ion Stoica, and Cho-Jui Hsieh. OR-Bench: An Over-Refusal Benchmark for Large Language Models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267. PMLR, 2025. URL https://proceedings.mlr.press/v267/cui25a.html.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. Plug and play language models: A simple approach to controlled text generation. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=H1edEyBKDS.

Jacob Dunefsky and Arman Cohan. One-shot optimized steering vectors mediate safety-relevant behaviors in LLMs. In *Second Conference on Language Modeling*, 2025. URL https://arxiv.org/abs/2502.18862.

Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37, pp. 1180–1189. PMLR, 2015. URL https://proceedings.mlr.press/v37/ganin15.html.

Navita Goyal and Hal Daumé III. Steering safely or off a cliff? rethinking specificity and robustness in inference-time interventions. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5723–5738, 2026.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Seungju Han, Kavel Rao, Allyson Ettinger, Liwei Jiang, Bill Yuchen Lin, Nathan Lambert, Yejin Choi, and Nouha Dziri. WildGuard: Open One-Stop Moderation Tools for Safety Risks, Jailbreaks, and Refusals of LLMs. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://papers.neurips.cc/paper_files/paper/2024/hash/0f69b4b96a46f284b726fbd70f74fb3b-Abstract-Datasets_and_Benchmarks_Track.html.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring Massive Multitask Language Understanding. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2009.03300.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=6t0Kwf8-jrj.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada, July 2017. Association for Computational Linguistics.

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, 2023. URL https://openreview.net/forum?id=aLLuYpn83y.

Yawei Li, Benjamin Bergner, Yinghan Zhao, Vihang Prakash Patil, Bei Chen, and Cheng Wang. Steering large reasoning models towards concise reasoning via flow matching. *Transactions on Machine Learning Research*, 2026. URL https://arxiv.org/abs/2602.05539.

Stephanie Lin, Jacob Hilton, and Owain Evans. Truthfulqa: Measuring how models mimic human falsehoods. In *Proceedings of the 60th annual meeting of the association for computational linguistics (volume 1: long papers)*, pp. 3214–3252, 2022.

Felipe Maia Polo, Lucas Weber, Leshem Choshen, Yuekai Sun, Gongjun Xu, and Mikhail Yurochkin. tinyBenchmarks: evaluating LLMs with fewer examples. *arXiv preprint arXiv:2402.14992*, 2024. URL https://arxiv.org/abs/2402.14992.

Mantas Mazeika, Long Phan, Xuwang Yin, Andy Zou, Zifan Wang, Norman Mu, Elham Sakhaee, Nathaniel Li, Steven Basart, Bo Li, David Forsyth, and Dan Hendrycks. HarmBench: A Standardized Evaluation Framework for Automated Red Teaming and Robust Refusal. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 35181–35224. PMLR, 2024. URL https://proceedings.mlr.press/v235/mazeika24a.html.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 12076–12100, 2023.

Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

Qwen Team. Qwen3.6-27B: Flagship-level coding in a 27B dense model, April 2026. URL https://qwen.ai/blog?id=qwen3.6-27b.

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, Xavier Suau, et al. Controlling language and diffusion models by transporting activations. In *International Conference on Learning Representations*, volume 2025, pp. 89812–89855, 2025.

Paul Röttger, Hannah Kirk, Bertie Vidgen, Giuseppe Attanasio, Federico Bianchi, and Dirk Hovy. Xstest: A test suite for identifying exaggerated safety behaviours in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 5377–5400, 2024.

Mrinank Sharma, Meg Tong, Tomek Korbak, David Duvenaud, Amanda Askell, Sam Bowman, Esin Durmus, Zac Hatfield-Dodds, Scott Johnston, Shauna Kravec, et al. Towards understanding sycophancy in language models. In *International Conference on Learning Representations*, volume 2024, pp. 110–144, 2024.

Daniel Tan, David Chanin, Aengus Lynch, Brooks Paige, Dimitrios Kanoulas, Adrià Garriga-Alonso, and Robert Kirk. Analysing the Generalisation and Reliability of Steering Vectors. In *Advances in Neural Information Processing Systems*, volume 37, pp. 139179–139212, 2024. URL https://discovery.ucl.ac.uk/id/eprint/10205543/.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

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.

Hieu M. Vu and Tan M. Nguyen. Angular Steering: Behavior Control via Rotation in Activation Space. In *Advances in Neural Information Processing Systems*, volume 38, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/b0223cad0e73b793f31eb6cc41cefceb-Abstract-Conference.html.

Qingsong Wang, Chang Yao, and Jingyuan Chen. Causal-steer: Disentangled continuous style control without parallel corpora. In *International Conference on Learning Representations*, volume 2026, pp. 32767–32788, 2026.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025a.

Songlin Yang, Jan Kautz, and Ali Hatamizadeh. Gated delta networks: Improving mamba2 with delta rule. In *International Conference on Learning Representations*, volume 2025, pp. 29687–29707, 2025b.

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*, 2023a. URL https://arxiv.org/abs/2310.01405.

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

Supplement to "COUNTERSTEER: Activation Steering from Undesirable Examples Alone"

Table of Contents

A Related Work **14**

B Notation **15**

C Algorithmic Details **15**

D Additional Experimental Details **16**

D.1 Common implementation details 16

D.2 Over-refusal steering details 17

D.3 Sycophancy steering details 18

E Analysis and Ablation Details **19**

E.1 Target ablation on Llama-3.1-8B-Instruct 20

E.2 Fixed-layer refusal–harmful-behavior trade-off 20

E.3 Inference Efficiency 21

E.4 Details of COUNTERSTEER-MOM 21

E.5 Results on Qwen3.6-27B 23

F Proofs and Scope of the Theoretical Analysis **23**

F.1 Proof of Proposition 1: local suppression 23

F.2 Normalized COUNTERSTEER-GRAD: construction and proof of Corollary 1 23

F.3 Adaptation-induced shifts and alignment 24

F.4 Transfer across inputs and token positions 25

G Use of Existing Assets **25**

A RELATED WORK

Activation Steering. Activation steering controls model behavior without parameter updates (Li et al., 2023). ActAdd uses activation differences between contrasting prompts (Turner et al., 2023), while CAA averages paired differences into reusable directions (Rimsky et al., 2024). Transport-based methods align activation distributions (Rodriguez et al., 2025); CHARS combines cluster-level shifts conditioned on the input (Abdullaev et al., 2026).

These constructions require examples from both sides of the intended behavioral change. COUNTERSTEER instead reverses a direction obtained from undesired-data fitting, without desired counterparts or a contrasting collection. This one-sided requirement concerns vector construction; layer and strength selection may still use labeled validation data.

Gradient-Based Control. Gradients have been used to shape model behavior through both training and inference-time interventions. Ganin & Lempitsky (2015) reverse domain-classification gradients to learn domain-invariant features. Plug and Play Language Models iteratively modify hidden states using attribute gradients (Dathathri et al., 2020).

Dunefsky & Cohan (2025) iteratively optimize steering vectors on individual examples, using token-level unlikelihood for suppression. Their one-shot setting refers to a single training example,

not a single optimization step. COUNTERSTEER-GRAD instead averages per-example normalized response-NLL gradients evaluated at the unsteered base model, avoiding vector initialization, optimizer hyperparameters, and stopping criteria. NLL ascent and unlikelihood descent generally yield different directions: the latter weights each token's NLL gradient by $p_t/(1-p_t)$, where p_t is the target-token probability under the same teacher-forced context, before summing over tokens. Normalization limits large-norm contributions, while aggregation combines evidence across examples. Our analysis characterizes local suppression of the matched fixed-weight score.

Reversing Adaptation-Induced Changes. Reversing model updates has been explored as a way to suppress learned behaviors. Task arithmetic negates fine-tuning displacements in parameter space to reduce targeted capabilities or undesirable behavior (Ilharco et al., 2023). Similarly, NEON improves image generation by extrapolating away from parameter updates induced by self-training on model-generated data (Alemohammad et al., 2025).

CAUSAL-STEER aggregates same-input adapted-minus-base activation shifts using PCA and a geometric median (Wang et al., 2026). Its single-source variant shares COUNTERSTEER-LORA's adaptation-and-reversal principle. COUNTERSTEER-LORA uses a simpler construction: it averages final-prompt-token shifts and reverses them at one decoder-block output, without PCA or geometric-median estimation. We study this variant alongside COUNTERSTEER-GRAD and characterize its alignment condition for local suppression.

B NOTATION

Table 5 summarizes the main notation used in the paper.

Table 5: **Main notation.**

Symbol	Description
$\mathcal{D}^-, \mathcal{D}^+$	Undesired and desired example collections.
$z_i^- = [x_i; y_i^-]$	Undesired example with prompt x_i and response y_i^-.
N_-	Number of undesired construction examples.
$\mathcal{T}_i, \mathcal{K}_i$	Scored token indices and intervention positions for example i.
$\ell, \boldsymbol{h}_t^{(\ell)}$	Layer index and its hidden state at token position t.
$\boldsymbol{\theta}_0, p_{\boldsymbol{\theta}_0}$	Frozen base-model parameters and predictive distribution.
$\boldsymbol{u}$	Shared activation offset at the intervened layer.
$\boldsymbol{v}^{(\ell)}, \alpha$	Steering vector and intervention strength.
$\bar{\boldsymbol{d}}^{(\ell)}$	Probed fitting direction, reversed as $\boldsymbol{v}^{(\ell)} = -\bar{\boldsymbol{d}}^{(\ell)}$.
$\mathcal{L}_i, \mathcal{L}_-$	Per-example masked mean NLL and its construction-set mean.
$\boldsymbol{q}_i^{(\ell)}$	Per-example loss gradient with respect to the shared offset.
$w_i^{(\ell)}$	Fixed Grad normalization weight $(\lVert \boldsymbol{q}_i^{(\ell)} \rVert_2 + \epsilon)^{-1}$.
$\boldsymbol{v}_{\text{Grad}}^{(\ell)}$	Steering vector constructed from activation gradients.
$\boldsymbol{\phi}_0, \boldsymbol{\phi}_T$	Adapter parameters before and after T fitting steps.
$\mathcal{X}_{\text{extract}}$	Inputs used to extract adapter-induced activation shifts.
$\boldsymbol{d}^{(\ell)}(x)$	Adapted-minus-base activation shift on input x.
$F(\alpha), F_w(\alpha)$	Unweighted and fixed-weight undesired-token log-likelihood scores.
$\boldsymbol{b}^{(\ell)}, m^{(\ell)}$	Base-model score gradient and first-order suppression margin.

C ALGORITHMIC DETAILS

The algorithms implement Section 2.3 and Section 2.4, returning construction vectors before final unit rescaling. Grad includes per-sequence normalization as in Eq. (8). Both vectors are deployed through Eq. (1), with the inference scaling specified separately for each experiment.

Algorithm 1 COUNTERSTEER-GRAD

Require: Frozen base model $p_{\boldsymbol{\theta}_0}$; undesired sequences $\{z_i^-\}_{i=1}^{N_-}$; scoring masks $\{\mathcal{T}_i\}$; intervention sites $\{\mathcal{K}_i\}$; layer ℓ; $\epsilon > 0$.

1: Set the model to evaluation mode and initialize $\boldsymbol{v}_{\text{Grad}}^{(\ell)} = \mathbf{0}$.
2: **for** each sequence z_i^- **do**
3: Evaluate the teacher-forced masked mean NLL $\mathcal{L}_i$ in Eq. (3).
4: Backpropagate to the layer-ℓ block outputs and compute $\boldsymbol{q}_i^{(\ell)} = \sum_{k \in \mathcal{K}_i} \nabla_{\boldsymbol{h}_{i,k}^{(\ell)}} \mathcal{L}_i$.
5: Accumulate $\boldsymbol{v}_{\text{Grad}}^{(\ell)} \leftarrow \boldsymbol{v}_{\text{Grad}}^{(\ell)} + \boldsymbol{q}_i^{(\ell)} / (\|\boldsymbol{q}_i^{(\ell)}\|_2 + \epsilon)$.
6: **end for**
7: Average: $\boldsymbol{v}_{\text{Grad}}^{(\ell)} \leftarrow \boldsymbol{v}_{\text{Grad}}^{(\ell)} / N_-$.

Ensure: Fixed reverse steering vector, with no parameter updates.

The weighted positive NLL gradient reverses the local descent direction of Eq. (9); no additional sign change is applied at inference. Each example is normalized before averaging, even when forward passes are batched. A zero example gradient contributes zero.

Algorithm 2 COUNTERSTEER-LORA

Require: Base model $p_{\boldsymbol{\theta}_0}$; undesired dataset $\mathcal{D}^-$ with loss positions $\{\mathcal{T}_i\}$; extraction inputs $\mathcal{X}_{\text{extract}}$; LoRA configuration and optimization budget T; layer ℓ.

1: Initialize adapter parameters $\boldsymbol{\phi}_0$ with zero effective weight update.
2: Optimize the mean masked NLL over $\mathcal{D}^-$, using Eq. (3) evaluated under the adapted model, for T steps with $\boldsymbol{\theta}_0$ frozen to obtain $\boldsymbol{\phi}_T$.
3: **for** each $x \in \mathcal{X}_{\text{extract}}$ **do**
4: Read $\boldsymbol{h}_{\boldsymbol{\phi}_0}^{(\ell)}(x)$ and $\boldsymbol{h}_{\boldsymbol{\phi}_T}^{(\ell)}(x)$ at the final non-padding input token in evaluation mode.
5: Compute $\boldsymbol{d}^{(\ell)}(x)$ using Eq. (11).
6: **end for**
7: Average the shifts to obtain $\bar{\boldsymbol{d}}^{(\ell)}$.
8: Reverse the mean shift: $\boldsymbol{v}^{(\ell)} = -\bar{\boldsymbol{d}}^{(\ell)}$.
9: Discard the adapter and return $\boldsymbol{v}^{(\ell)}$.

Ensure: Fixed reverse steering vector for the original base model.

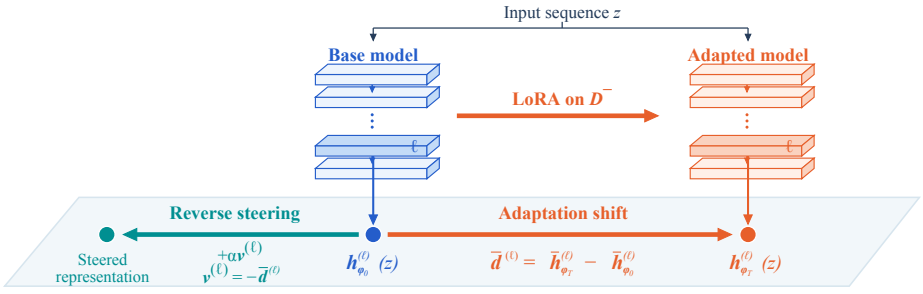


Figure 5: **Overview of the COUNTERSTEER-LORA variant.** LoRA adaptation on undesired examples induces a representation shift, measured by comparing the adapted and base models on identical extraction inputs. Averaging these shifts gives $\bar{\boldsymbol{d}}^{(\ell)}$; its reverse $\boldsymbol{v}^{(\ell)} = -\bar{\boldsymbol{d}}^{(\ell)}$ steers the base model with the adapter disabled. The plane is schematic; the mean shift is computed over the extraction collection.

D ADDITIONAL EXPERIMENTAL DETAILS

D.1 COMMON IMPLEMENTATION DETAILS

Runtime and generation. The following settings apply to the main additive-steering experiments. We use Hugging Face Transformers and PEFT with BF16 precision. Evaluation uses greedy decod-

Table 6: **LoRA training settings.**

LoRA setting	Value
Rank / scaling	16 / 32
Optimizer / effective batch size	AdamW / 16
Dropout / gradient clipping norm	0 / 1
Default learning rate	10^{-4}
Over-refusal learning-rate exceptions	Llama 5×10^{-6}, Gemma 10^{-5}

Table 7: **Over-refusal data splits.** Counts refer to prompts.

Role	Dataset	Count
CounterSteer construction	OR-Bench-80k	512
LoRA extraction	OR-Bench-80k	128
Benign validation	OR-Hard	256
Harmful validation	OR-Toxic	256
Test	OR-Hard	1,000
Test	XSTest (benign)	250
Test	HarmBench	300
Test	OR-Toxic	399

ing, with up to 512 new tokens for free-form responses and 8 for short answer choices. Prompts follow each model's chat format, preserving existing conversation histories.

Vector construction and intervention. Layer indices are zero-based decoder-block outputs. Grad uses evaluation mode and per-sequence gradient normalization with $\epsilon = 10^{-8}$. Separately, we unit-normalize each final steering vector and apply it at one layer to all prompt and generated tokens, without hidden-state-norm scaling. DiM and LAT-PCA use the same task-specific contrast activations. LAT-PCA takes the first principal component of centered pair differences after randomizing within-pair order. Contrast labels determine its sign, and all directions are oriented to suppress the target behavior.

LoRA training. LoRA updates attention and MLP projection matrices while keeping base weights frozen. Training uses response-only mean token NLL, averaged across examples. Checkpoints are selected using a fixed task-specific gate that checks target behavior and output quality, without using test results. Table 6 summarizes the training hyperparameters.

Configuration selection. We search the latter half of decoder layers, select layer and strength on validation data, and freeze them for testing. The one-sided data requirement concerns CounterSteer vector construction, not validation or evaluation.

D.2 Over-refusal steering details

Dataset sourcing. We pair benign OR-Bench-80k prompts with fixed refusal templates and reserve separate prompts for LoRA extraction. Targets therefore need not be refusals generated by the base model. Normalized-text deduplication excludes test prompts from construction and validation. Table 7 lists the datasets and split sizes. The loaded `or-bench-hard-1k` subset contains 1,319 rows despite its name. After deduplication and exclusion of overlaps with the external evaluation sets, we shuffle the remaining prompts with seed 42 and allocate 256 to validation and a separate 1,000 to testing. The saved splits have no overlap under NFKC normalization, case folding, and whitespace normalization.

LoRA extraction. For CounterSteer-LoRA, refusal responses are used to fit the adapter, whereas the 128 extraction inputs are formatted prompts without response targets. We feed identical inputs to the base and adapted models, read the decoder-block output at the final non-padding token, reverse the mean shift, and discard the adapter before evaluation (Algorithm 2).

Baseline extraction. DiM contrasts final-prompt-token activations from 128 harmful AdvBench prompts and 128 benign OR-Bench prompts. LAT-PCA uses the same contrast set. These baselines require harmful and benign prompt collections, but no helpful completions. CounterSteer

Table 8: **Selected over-refusal configurations.** Each entry $L = xx,\ \alpha = yy$ denotes zero-based layer xx and steering strength $\alpha = yy$.

Model	DiM	LAT-PCA	CounterSteer-Grad	CounterSteer-LoRA
Qwen2.5-3B-Instruct	$L = 20,\ \alpha = 2.0$	$L = 18,\ \alpha = 4.0$	$L = 26,\ \alpha = 8.0$	$L = 22,\ \alpha = 4.0$
Qwen2.5-7B-Instruct	$L = 23,\ \alpha = 16.0$	$L = 14,\ \alpha = 1.0$	$L = 27,\ \alpha = 8.0$	$L = 25,\ \alpha = 8.0$
Llama-3.1-8B-Instruct	$L = 19,\ \alpha = 1.0$	$L = 30,\ \alpha = 4.0$	$L = 27,\ \alpha = 2.0$	$L = 17,\ \alpha = 2.0$
Gemma-2-9B-it	$L = 21,\ \alpha = 32.0$	$L = 21,\ \alpha = 32.0$	$L = 22,\ \alpha = 128.0$	$L = 23,\ \alpha = 8.0$

Table 9: **Sycophancy data splits.** Probe counts are response rows, not necessarily unique items. Extraction counts are prompt-only inputs. Gate, validation, and test counts are questions or feedback texts before expansion into user framings. Gate data are used for LoRA checkpoint selection.

Form	Probe responses	LoRA extraction	Gate	Validation	Test
Answer agreement	171	43	128	256	740
Pushback	171	43	128	150	300
Feedback	170	43	128	100	300

instead constructs its vectors from benign prompts paired with undesired refusals, without a contrasting prompt collection.

Configuration selection. We maximize validation benign non-refusal while allowing at most a two-percentage-point increase in harmful-response rate over the unsteered model. This allowance constrains validation selection and does not guarantee the same bound on held-out benchmarks. Table 8 reports the selected layer and strength for each model and method.

Metrics. WildGuard judges refusal on OR-Hard and benign XSTest, while HarmBench-Mistral-7b-val-cls judges harmful responses on HarmBench and OR-Toxic. Non-refusal does not imply correctness or helpfulness, and harmful-response rate is not the complement of refusal rate. For benchmark rate r_b and sample count n_b, the macro average is $\frac{1}{4}\sum_b r_b$, while the micro average is $\sum_b n_b r_b / \sum_b n_b$, using counts $(1000, 250, 300, 399)$ in table order. Neither average includes capability accuracy. We evaluate 100 items from each tinyBenchmarks task—MMLU, ARC-Challenge, HellaSwag, WinoGrande, TruthfulQA, and GSM8K—using the same steering configurations without retuning, and report their mean accuracy.

D.3 Sycophancy steering details

Dataset sourcing. Each model supplies 512 undesired responses across three forms. Answer construction uses TriviaQA validation questions with generated, checked wrong suggestions. Pushback construction and validation use MMLU examples from the *Are You Sure?* data, while testing uses TruthfulQA-MC. Feedback uses arguments and poems from sycophancy-eval. Test questions and texts are excluded from probe construction and selection. Table 9 gives the split sizes for each form.

Baseline extraction. Response-contrast DiM reads the final response token from 512 pairs with the same form quotas. LAT-PCA uses these pairs. Contrast responses are correct answers, retained answers after pushback, or feedback that does not follow the user's preference. When unavailable, pushback uses the original correct answer letter and feedback uses the model's neutral-framing response. Thus, some contrasts are constructed rather than sampled under the same framing.

Configuration selection. Validation maximizes the mean relative reduction across the three forms, subject to accuracy and output-quality guards. Feedback selection uses a rubric-based rating spread, whereas testing uses pairwise positivity judgments. Table 10 reports the selected layer and strength for each model and method. The random control uses a normalized Gaussian direction at LoRA's selected layer and strength, without a separate search.

Metrics. Qwen3-32B judges answer commitment and feedback positivity.

- **Sycophantic agreement:** fraction of responses endorsing the user's incorrect suggestion.
- **Correction:** fraction giving the correct answer despite the incorrect suggestion.

Table 10: **Selected sycophancy configurations.** Each entry $L = xx,\ \alpha = yy$ denotes zero-based layer xx and steering strength $\alpha = yy$.

Model	DiM	LAT-PCA	CounterSteer-Grad	CounterSteer-LoRA
Qwen2.5-3B-Instruct	$L = 33,\ \alpha = 64.0$	$L = 33,\ \alpha = 32.0$	$L = 20,\ \alpha = 4.0$	$L = 28,\ \alpha = 32.0$
Qwen2.5-7B-Instruct	$L = 17,\ \alpha = 16.0$	$L = 17,\ \alpha = 16.0$	$L = 14,\ \alpha = 4.0$	$L = 20,\ \alpha = 32.0$
Llama-3.1-8B-Instruct	$L = 25,\ \alpha = 1.0$	$L = 26,\ \alpha = 2.0$	$L = 16,\ \alpha = 1.0$	$L = 16,\ \alpha = 1.0$

- **Flip rate:** fraction of each method's initially correct answers that become incorrect after pushback. The denominator and evaluated subset may therefore differ across methods; this conditional rate does not measure overall post-pushback accuracy.
- **Feedback positivity gap:** like-minus-dislike difference in the probability of being judged more positive than the same arm's neutral response, reported in percentage points (pp). Both comparison orders are judged.

E Analysis and Ablation Details

Construction normalization and inference scaling.

The per-sequence normalization in Eq. (8) is part of the CounterSteer-Grad construction. It is distinct from rescaling the final vector for an experiment. In the unit-vector comparisons below, a nonzero construction vector $\boldsymbol{v}^{(\ell)}$ is replaced by $\widehat{\boldsymbol{v}}^{(\ell)} = \boldsymbol{v}^{(\ell)}/\|\boldsymbol{v}^{(\ell)}\|_2$, and inference uses $\boldsymbol{h} + \alpha_{\text{unit}}\widehat{\boldsymbol{v}}^{(\ell)}$. This equals the unnormalized intervention when $\alpha_{\text{unit}} = \alpha_{\text{raw}}\|\boldsymbol{v}^{(\ell)}\|_2$. A zero construction vector produces no intervention. No hidden-state-norm scaling is applied in these comparisons. Positive rescaling preserves direction and local alignment, but changes the numerical strength and curvature along the intervention; strengths using different conventions must not be compared directly. The corresponding Grad derivative is worked out in Appendix F.2.

Configurations of the reported analyses.

The alignment and target ablations use 102 audited prompt/target pairs, with zero-based layers 14–27 for Qwen and 16–31 for Llama. The trade-off study uses the main experiment's 512 training pairs. All Grad directions follow Eq. (8); all interventions use unit-normalized vectors as defined above. For cosine comparisons, all vectors are oriented toward non-refusal. Cosine similarity to DiM describes directional similarity, not the score-gradient alignment required by the local suppression analysis in Section 3. It therefore does not by itself verify that condition. The DiM direction is the negative of the harmful-minus-harmless prompt activation difference, read at the final non-padding token; the same orientation is used in the behavioral sweeps.

Target ablation on Qwen2.5-7B-Instruct.

Both Grad and LoRA target comparisons use the same 102 audited prompts and fixed refusal/positive completion pairs. Positive targets were generated by a fixed Qwen2.5-14B-Instruct teacher and assistant-reviewed, with 26 of 102 responses corrected; they are not human-validated reference answers. Differences in target content, length, and style remain part of this construction diagnostic. For Grad, response-only mean NLL gradients are summed over non-padding intervention sites, divided by each sequence gradient's L2 norm plus 10^{-8}, and averaged across examples. The refusal condition uses positive NLL gradients; the positive-target condition negates its NLL gradients. Extraction uses the frozen model in evaluation mode, with no optimizer steps. Tokenization and EOS handling are shared across the two target conditions, with no truncation. Both LoRA adapters retain the same hyperparameters, seed, and two-epoch optimization budget.

All behavioral curves use the same 64 benign validation prompts, greedy decoding, and a 512-token generation limit. Non-refusal is the fraction of valid outputs assigned `response_refusal=false` by the WildGuard (Han et al., 2024) native classifier used in the main evaluation; it does not measure correctness or substantive helpfulness. All reported conditions have 64 valid judgments and zero invalid ones. Directions are unit-normalized, without activation-norm scaling, and applied at one decoder-block output to prompt and generated tokens. The common $\alpha = 8$ for layers 14–27 was selected in the original LoRA comparison to maximize CounterSteer-LoRA's mean validation non-refusal over $\{0.5, 1, 2, 4, 8\}$; it was not retuned using Grad outcomes. The base-model non-refusal rate is 82.81%. These curves do not apply the harmful-behavior constraint used for the main test-set operating points.

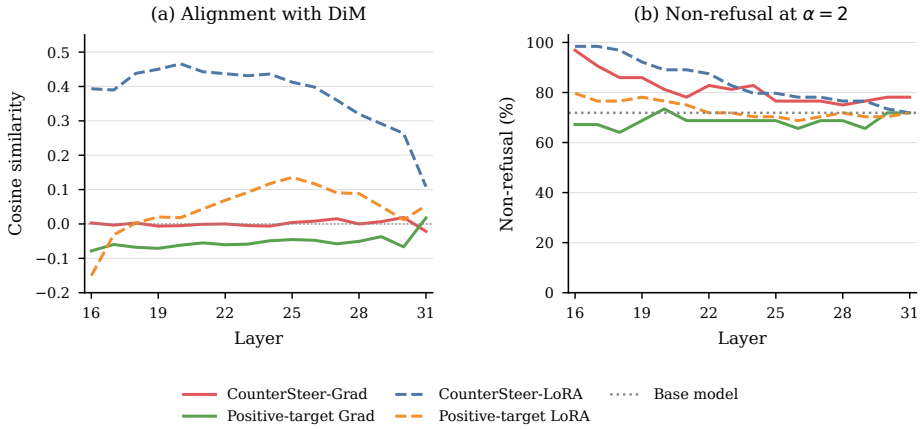


Figure 6: **Target ablation on Llama-3.1-8B-Instruct. (a)** Alignment with DiM. **(b)** Validation non-refusal at $\alpha = 2$. Solid: Grad; dashed: LoRA; dotted: base model.

ADAPTER-ON DIAGNOSTIC.

The original positive-target adapter-on control attains 93.75% non-refusal on Qwen, compared with the base model's 82.81%. It changes all trained modules, whereas the extracted steering vector acts at one layer. It is therefore a mechanism diagnostic rather than a matched-budget steering baseline, and is omitted from the target comparison plot for clarity.

E.1 TARGET ABLATION ON LLAMA-3.1-8B-INSTRUCT

We repeat both target comparisons on Llama-3.1-8B-Instruct using the same 102 prompt/target pairs and 64 validation prompts. Grad follows the same normalization and sign conventions as above. Both LoRA adapters are trained afresh with rank 16, LoRA scaling 32, learning rate 10^{-4}, effective batch size 16, seed 0, and two epochs. Directions are extracted at zero-based layers 16–31. Figure 6 uses a fixed common $\alpha = 2$, without retuning for Grad; it does not replace the main DiM operating point ($\alpha = 1$, layer 19).

Figure 6 shows that reversal outperforms positive-target Grad at every evaluated layer and positive-target LoRA at all but one tied layer. Despite near-zero alignment with DiM, COUNTERSTEER-GRAD improves non-refusal over the base model throughout the sweep. Strong positive DiM alignment is therefore not necessary for these improvements. The positive adapter-on control reaches 98.44% non-refusal, but changes all trained modules rather than applying a single-layer vector. As with Qwen, these are fixed-strength diagnostics, not independently tuned or safety-constrained comparisons.

E.2 FIXED-LAYER REFUSAL–HARMFUL-BEHAVIOR TRADE-OFF

For Figure 4, we use the main-experiment Qwen2.5-7B-Instruct directions for COUNTERSTEER-LORA and DiM. COUNTERSTEER-GRAD is constructed on the 512 negative training pairs from the main setup, using the frozen base model, response-only mean token NLL per example, and no parameter update. Activation gradients are summed over non-padding intervention positions, divided per sequence by their L2 norm plus 10^{-8}, and then averaged, as in Eq. (8). All three methods use zero-based layer 23 and $\alpha \in \{0, 4, 8, 12, 16, 24, 32\}$, with $\alpha = 0$ sharing the base-model result; labels in the figure mark $\alpha = 16$ and 32. We normalize each direction and do not multiply the intervention by the hidden-state norm. Layer 23 is fixed for this comparison and is not claimed to be optimal for COUNTERSTEER-GRAD. No layer or strength is selected using these test curves.

The vertical axis is the arithmetic mean of WildGuard (Han et al., 2024) refusal rates on OR-Hard (1,000 prompts) and benign XSTest (250 prompts). The horizontal axis is the arithmetic mean of harmful-behavior rates on HarmBench (300 prompts) and the held-out OR-Toxic split (399 prompts), using the same HarmBench Mistral validation classifier (Mazeika et al., 2024) as the main evaluation. Each benchmark receives equal weight, rather than pooling samples across datasets. The target model uses the same pinned revision, BF16 SDPA, greedy decoding, and 512-token limit

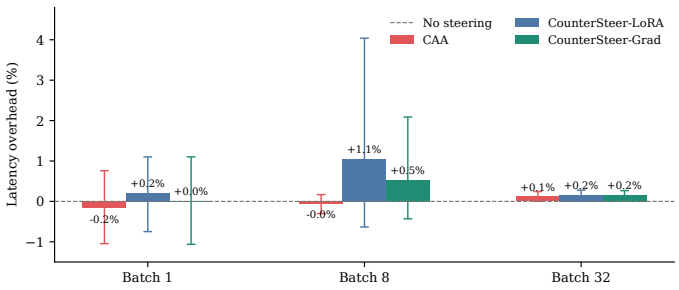


Figure 7: **Inference latency overhead relative to no steering.** All steering methods use the same single-layer additive intervention at inference. Error bars show paired bootstrap 95% confidence intervals over 12 interleaved measurements.

Table 11: **Momentum ablation on Qwen2.5-3B-Instruct.** Sycophancy metrics follow Table 3.

Method	Answer Sycophancy		Feedback Sycophancy	Are You Sure?
	Syc. Agreement (%) ↓	Correction (%) ↑	Positivity Gap (pp) ↓	Flip (%) ↓
Original (no steer)	34.7	46.5	78.1	37.5
DiM	23.5	50.8	83.6	42.2
LAT-PCA	25.9	49.1	81.6	39.8
CounterSteer-Grad (Ours)	23.1	**52.8**	73.7	**4.4**
CounterSteer-LoRA (Ours)	**22.0**	49.2	**64.9**	11.9
CounterSteer-Mom *(Ours)*	*25.5*	*50.3*	*79.7*	*12.9*

as the main experiment. All conditions have zero invalid judgments and zero empty responses. The connected points are descriptive operating curves, not confidence bands or evidence of uniform Pareto dominance. The one-sided requirement concerns direction construction: model selection and evaluation still use labeled benign and harmful data.

E.3 Inference Efficiency

We benchmark inference latency on Qwen2.5-7B-Instruct using one NVIDIA H100 80GB GPU with BF16 SDPA and KV caching. Inputs are padded to 128 tokens and each sequence generates exactly 128 new tokens. CAA, CounterSteer-LoRA, and CounterSteer-Grad use unit-normalized vectors at layer 20 with $\alpha = 8$. Model loading, tokenization, and vector construction are excluded. This benchmark measures the shared additive inference operation; its latency result does not depend on how the fixed direction was constructed. We use two warm-up passes followed by 12 interleaved measurements per method and batch size. Figure 7 reports mean overhead relative to generation without steering; error bars are paired bootstrap 95% confidence intervals. The observed mean overhead is at most 1.1% in this setting.

E.4 Details of CounterSteer-Mom

Table 11 reports the momentum ablation on Qwen2.5-3B-Instruct using the sycophancy metrics from Table 3. We detail the construction below.

Gradient construction.

For a chosen decoder layer ℓ, let $u \in \mathbb{R}^d$ be a shared offset added to the block output at every non-padding intervention position. The model weights remain frozen. For each undesired sequence i, let $\mathcal{L}_i(u)$ be the mean negative log-likelihood over its response tokens, evaluated with teacher forcing. Prompt and padding tokens are excluded from the loss, but prompt positions can still contribute activation gradients.

Algorithm 3 COUNTERSTEER-MOM

Require: Frozen base model $p_{\boldsymbol{\theta}_0}$; undesired sequences $\{z_i^-\}_{i=1}^{N_-}$; scoring masks $\{\mathcal{T}_i\}$; intervention sites $\{\mathcal{K}_i\}$; layer ℓ; nonempty index batches $\{B_t\}_{t=0}^{T-1}$; $T \geq 1$; $0 \leq \beta < 1$; $\eta > 0$; $\epsilon > 0$.

1: Set the model to evaluation mode and initialize $\boldsymbol{u}_0 = \boldsymbol{m}_0^{(\ell)} = \mathbf{0}$.
2: **for** $t = 0, \ldots, T-1$ **do**
3: Initialize $\boldsymbol{g}_t^{(\ell)} = \mathbf{0}$ and hold $\boldsymbol{u}_t$ fixed throughout this step.
4: **for** each sequence index $i \in B_t$ **do**
5: Add the shared offset $\boldsymbol{u}_t$ to the layer-ℓ block outputs at positions $k \in \mathcal{K}_i$.
6: Evaluate the teacher-forced mean NLL $\mathcal{L}_i(\boldsymbol{u}_t)$ over tokens in $\mathcal{T}_i$.
7: Backpropagate to the offset block outputs and compute $\boldsymbol{q}_{i,t}^{(\ell)} = \sum_{k \in \mathcal{K}_i} \nabla_{\widetilde{\boldsymbol{h}}_{i,k}^{(\ell)}} \mathcal{L}_i(\boldsymbol{u}_t)$.
8: Accumulate $\boldsymbol{g}_t^{(\ell)} \leftarrow \boldsymbol{g}_t^{(\ell)} + \boldsymbol{q}_{i,t}^{(\ell)} / (\|\boldsymbol{q}_{i,t}^{(\ell)}\|_2 + \epsilon)$.
9: **end for**
10: Average: $\boldsymbol{g}_t^{(\ell)} \leftarrow \boldsymbol{g}_t^{(\ell)} / |B_t|$.
11: Update momentum: $\boldsymbol{m}_{t+1}^{(\ell)} \leftarrow \beta \boldsymbol{m}_t^{(\ell)} + (1-\beta)\boldsymbol{g}_t^{(\ell)}$.
12: Update offset: $\boldsymbol{u}_{t+1} \leftarrow \boldsymbol{u}_t - \eta \boldsymbol{m}_{t+1}^{(\ell)}$.
13: Detach the updated offset and momentum from the computation graph.
14: **end for**
15: Set $\boldsymbol{v}_{\mathrm{Mom}}^{(\ell)} \leftarrow \boldsymbol{m}_T^{(\ell)} / (1-\beta^T)$ and discard $\boldsymbol{u}_T$.

Ensure: Fixed reverse steering vector $\boldsymbol{v}_{\mathrm{Mom}}^{(\ell)}$, with no model-parameter updates.

At step t, we sum the activation gradients over the intervention positions K_i, normalize each sequence's gradient, and average over the examples B_t used at that step:

$$q_{i,t}^{(\ell)} = \nabla_u \mathcal{L}_i(u)|_{u=u_t} = \sum_{k \in K_i} \nabla_{\widetilde{h}_{i,k}^{(\ell)}} \mathcal{L}_i(u_t), \tag{21}$$

$$g_t^{(\ell)} = \frac{1}{|B_t|} \sum_{i \in B_t} \frac{q_{i,t}^{(\ell)}}{\|q_{i,t}^{(\ell)}\|_2 + \epsilon}. \tag{22}$$

Here $\epsilon > 0$ prevents division by zero. Normalization is applied after summing over positions, not separately to each token or to the batch mean. The gradients are recomputed at the current offset; the procedure does not repeatedly average the same base-model gradient.

MOMENTUM UPDATES AND STEERING.

Starting from $u_0 = m_0^{(\ell)} = 0$, we perform T updates:

$$m_{t+1}^{(\ell)} = \beta m_t^{(\ell)} + (1-\beta) g_t^{(\ell)}, \tag{23}$$

$$u_{t+1} = u_t - \eta m_{t+1}^{(\ell)}, \qquad t = 0, \ldots, T-1, \tag{24}$$

where $0 \leq \beta < 1$ and $\eta > 0$. The offset follows the negative momentum direction to fit undesired responses. To reverse this direction, we define

$$v_{\mathrm{Mom}}^{(\ell)} = \frac{m_T^{(\ell)}}{1-\beta^T} = \frac{1-\beta}{1-\beta^T} \sum_{t=0}^{T-1} \beta^{T-1-t} g_t^{(\ell)}. \tag{25}$$

We discard u_T and apply the normalized vector to the original model:

$$\widetilde{h}_k^{(\ell)} = h_k^{(\ell)} + \alpha \frac{v_{\mathrm{Mom}}^{(\ell)}}{\|v_{\mathrm{Mom}}^{(\ell)}\|_2}. \tag{26}$$

The positive sign reverses the fitting update; no further sign change is needed. The construction step size η and inference strength α are separate parameters. The bias correction in the final vector is a positive scalar, so it does not affect its unit direction. If the final vector is zero, we apply no intervention.

Each sequence gradient is computed with the offset held fixed within the step. Updates are detached from the computation graph; no gradient is taken through the optimization history. Reject non-finite gradients or vectors rather than normalizing them.

Table 12: **Over-refusal steering on Qwen3.6-27B.** Metrics and averaging conventions follow Table 1.

Method	Benign Refusal ↓		Harmful Behavior ↓		Average ↓	
	OR-Hard (%)	XSTest (%)	HarmBench (%)	OR-Toxic (%)	Macro avg (%) ↓	Micro avg (%) ↓
Original (no steer)	91.1	7.2	**1.3**	**0.50**	25.0	47.98
DiM	38.6	**0.8**	7.0	4.76	12.79	21.97
CounterSteer-Grad (Ours)	37.2	**0.8**	10.3	4.26	13.15	21.65
CounterSteer-LoRA (Ours)	**30.1**	**0.8**	7.7	3.5	**10.53**	**17.45**

Relation to CounterSteer-Grad.

With $T = 1$, $v^{(\ell)}_{\text{Mom}} = g^{(\ell)}_0$. This recovers CounterSteer-Grad when the first step uses the same construction examples, masks, intervention positions, and normalization. A single minibatch step recovers only the Grad vector for that minibatch. For $T > 1$, gradients are evaluated at nonzero offsets, so their average need not align with the base-model gradient. The local suppression guarantee for Grad's matched, fixed-weight construction score therefore does not automatically extend to Mom. This is one possible explanation for weaker steering, not a mechanism established by the reported results. A comparison with multi-step updates at $\beta = 0$ would be needed to separate the effect of momentum from taking additional steps.

E.5 Results on Qwen3.6-27B

Table 12 reports benign refusal and harmful-response rates on Qwen3.6-27B, extending the over-refusal evaluation to a newer, larger hybrid model. Metrics and averaging conventions follow the main experiments.

F Proofs and Scope of the Theoretical Analysis

We follow Section 3: first the common local suppression bound, then normalized CounterSteer-Grad, and finally the alignment required for CounterSteer-LoRA. The transfer bound extends this analysis. Throughout, sequences, scoring masks, intervention sites, and constructed vectors are fixed under teacher forcing. Activation derivatives are computed at the unsteered base model; adapter derivatives are evaluated at the stated adapter parameters. The guarantees concern differentiable scores on fixed sequences, not sampled responses.

F.1 Proof of Proposition 1: local suppression

For each $z \in \mathcal{E}^-$, define the shared-offset loss $\widetilde{\mathcal{L}}_z(\boldsymbol{u}) := \mathcal{L}_z(p^{(\ell)}_{\boldsymbol{\theta}_0,\boldsymbol{u}})$ using the scoring mask and intervention sites of Section 3. Since $S_z(\alpha) = -\widetilde{\mathcal{L}}_z(\alpha \boldsymbol{v}^{(\ell)})$, the chain rule gives

$$S'_z(0) = -\langle \nabla_{\boldsymbol{u}} \widetilde{\mathcal{L}}_z(\mathbf{0}), \boldsymbol{v}^{(\ell)} \rangle.$$

Averaging over $\mathcal{E}^-$ proves Eq. (14). The shared-offset derivative includes all intervention sites and their downstream effects, as in Eq. (5).

Taylor's theorem with an integral remainder gives

$$\begin{aligned} F(\alpha) - F(0) &= \alpha F'(0) + \int_0^\alpha (\alpha - s) F''(s)\, \mathrm{d}s \\ &\le -\alpha m^{(\ell)} + \frac{K_F}{2}\alpha^2, \qquad 0 \le \alpha \le a. \end{aligned}$$

For $m^{(\ell)} > 0$, the right-hand side is negative for the range in Proposition 1. It is at most $-\alpha m^{(\ell)}/2$ when additionally $\alpha \le m^{(\ell)}/K_F$. If $m^{(\ell)} = 0$, the first-order term is inconclusive; if $m^{(\ell)} < 0$, the score increases for sufficiently small positive α. A zero intervention vector leaves the score unchanged.

F.2 Normalized CounterSteer-Grad: construction and proof of Corollary 1

Construction convention.

We use Eqs. (5)–(9) and Algorithm 1: sum activation gradients over intervention sites, normalize each sequence vector, then average. The weights are fixed at the unsteered base model, and the aggregate is not necessarily a unit vector.

PROOF OF COROLLARY 1.

With the weights held fixed, Eq. (9) implies

$$\nabla_{\boldsymbol{u}} \mathcal{L}^{(\ell)}_{-,w}(\mathbf{0}) = \frac{1}{N_-} \sum_i w_i^{(\ell)} \boldsymbol{q}_i^{(\ell)} = \boldsymbol{v}^{(\ell)}_{\mathrm{Grad}}.$$

For the matched score in Eq. (16),

$$F'_w(0) = -\left\langle \nabla_{\boldsymbol{u}} \mathcal{L}^{(\ell)}_{-,w}(\mathbf{0}), \boldsymbol{v}^{(\ell)}_{\mathrm{Grad}} \right\rangle = -\|\boldsymbol{v}^{(\ell)}_{\mathrm{Grad}}\|_2^2.$$

No derivative passes through the weights. Applying the integral-remainder argument above to F_w gives

$$F_w(\alpha) - F_w(0) \le -\alpha \|\boldsymbol{v}^{(\ell)}_{\mathrm{Grad}}\|_2^2 + \frac{K_w}{2}\alpha^2, \qquad 0 \le \alpha \le a_w.$$

For a nonzero vector, this is strictly negative in the range stated in the corollary. If the aggregate is zero, no intervention is applied.

FINAL UNIT-VECTOR RESCALING.

Per-sequence normalization does not make the aggregate a unit vector. Let $c = \|\boldsymbol{v}^{(\ell)}_{\mathrm{Grad}}\|_2 > 0$ and $\widehat{\boldsymbol{v}}^{(\ell)} = \boldsymbol{v}^{(\ell)}_{\mathrm{Grad}}/c$. For a strength β applied to this unit vector, define $\widehat{F}_w(\beta) := -\mathcal{L}^{(\ell)}_{-,w}(\beta \widehat{\boldsymbol{v}}^{(\ell)})$. Then

$$\widehat{F}_w(\beta) = F_w(\beta/c), \qquad \widehat{F}'_w(0) = -c, \qquad |\widehat{F}''_w(\beta)| \le K_w/c^2 \quad (0 \le \beta \le c a_w).$$

Hence

$$\widehat{F}_w(\beta) - \widehat{F}_w(0) \le -\beta c + \frac{K_w}{2c^2}\beta^2,$$

which is negative for $0 < \beta < \min\{c a_w, 2c^3/K_w\}$. The strengths satisfy $\beta = c\alpha$, matching $\alpha_{\mathrm{unit}} = c\alpha_{\mathrm{raw}}$ in Appendix E; "raw" here means before final rescaling, not before per-sequence normalization. The bound is a local sufficient condition, not a practical strength-selection rule.

SCOPE OF THE GUARANTEE.

For the unweighted score on any evaluation collection $\mathcal{E}^-$, Eq. (14) instead gives the margin

$$m^{(\ell)}_{\mathcal{E}^-,\mathrm{Grad}} = -\langle \boldsymbol{b}^{(\ell)}_{\mathcal{E}^-}, \boldsymbol{v}^{(\ell)}_{\mathrm{Grad}} \rangle,$$

where $\boldsymbol{b}^{(\ell)}_{\mathcal{E}^-}$ is the unweighted aggregate in that equation. This margin need not be positive, even on the construction collection. The squared-norm identity requires the same examples, scoring masks, intervention sites, and fixed weights as construction. Changing any of these requires checking alignment anew. The local guarantee for F_w does not imply improved free-generation metrics or preservation of a reference task.

F.3 ADAPTATION-INDUCED SHIFTS AND ALIGNMENT

Use the adapter displacement $\Delta\boldsymbol{\phi} = \boldsymbol{\phi}_T - \boldsymbol{\phi}_0$ and extraction inputs $\{\bar{x}_j\}_{j=1}^M$ from Section 3.2. Let $\boldsymbol{J}^{(\ell)}_{\boldsymbol{\phi}}(x) \in \mathbb{R}^{d\times p}$ be the Jacobian of $\boldsymbol{h}^{(\ell)}_{\boldsymbol{\phi}}(x)$ with respect to the adapter parameters $\boldsymbol{\phi} \in \mathbb{R}^p$. Assume continuous differentiability near the segment joining the two adapter states and an $L^{(\ell)}_x$-Lipschitz Jacobian along it, in operator norm relative to the Euclidean parameter norm. These are the regularity assumptions of the main text, not consequences of the LoRA parameterization.

For the endpoint shift $\boldsymbol{d}^{(\ell)}(x)$ in Eq. (11), Taylor's theorem gives

$$\boldsymbol{d}^{(\ell)}(x) = \boldsymbol{J}^{(\ell)}_{\boldsymbol{\phi}_0}(x)\Delta\boldsymbol{\phi} + \boldsymbol{r}^{(\ell)}(x), \qquad \|\boldsymbol{r}^{(\ell)}(x)\|_2 \le \frac{L^{(\ell)}_x}{2}\|\Delta\boldsymbol{\phi}\|_2^2. \tag{27}$$

Here $\boldsymbol{r}^{(\ell)}(x)$ is the difference between the actual shift and its first-order approximation. The bound follows by integrating the Jacobian difference along the segment: at fractional distance $s \in [0,1]$, its contribution is bounded by $L^{(\ell)}_x s\|\Delta\boldsymbol{\phi}\|_2^2$.

Averaging over extraction inputs yields Eq. (19), with $\bar{\boldsymbol{J}}^{(\ell)} = M^{-1}\sum_j \boldsymbol{J}^{(\ell)}_{\boldsymbol{\phi}_0}(\bar{x}_j)$, $\bar{\boldsymbol{r}}^{(\ell)} = M^{-1}\sum_j \boldsymbol{r}^{(\ell)}(\bar{x}_j)$, and $\bar{L}^{(\ell)} = M^{-1}\sum_j L^{(\ell)}_{\bar{x}_j}$. The triangle inequality gives $\|\bar{\boldsymbol{r}}^{(\ell)}\|_2 \le \bar{L}^{(\ell)}\|\Delta\boldsymbol{\phi}\|_2^2/2$. Since $\boldsymbol{v}^{(\ell)} = -\bar{\boldsymbol{d}}^{(\ell)}$, Cauchy–Schwarz then gives

$$\begin{aligned} m^{(\ell)} &= \langle \boldsymbol{b}^{(\ell)}, \bar{\boldsymbol{J}}^{(\ell)}\Delta\boldsymbol{\phi}\rangle + \langle \boldsymbol{b}^{(\ell)}, \bar{\boldsymbol{r}}^{(\ell)}\rangle \\ &\ge \langle \boldsymbol{b}^{(\ell)}, \bar{\boldsymbol{J}}^{(\ell)}\Delta\boldsymbol{\phi}\rangle - \frac{\bar{L}^{(\ell)}}{2}\|\boldsymbol{b}^{(\ell)}\|_2\|\Delta\boldsymbol{\phi}\|_2^2. \end{aligned}$$

Table 13: **Model checkpoints, experimental roles, and licenses.**

Asset	Use in this work	License / terms
`Qwen/Qwen2.5-3B-Instruct` (Qwen et al., 2025)	Target model	Qwen Research License
`Qwen/Qwen2.5-7B-Instruct` (Qwen et al., 2025)	Target model	Apache-2.0
`meta-llama/Llama-3.1-8B-Instruct` (Grattafiori et al., 2024)	Target model	Llama 3.1 Community License
`google/gemma-2-9b-it` (Team et al., 2024)	Target model	Gemma Terms of Use
`Qwen/Qwen3.6-27B`(Qwen Team, 2026)	Target model (larger-model ablation)	Apache-2.0
`Qwen/Qwen2.5-14B-Instruct` (Qwen et al., 2025)	Positive-target generation (training ablation)	Apache-2.0
`Qwen/Qwen3-32B` (Yang et al., 2025a)	Sycophancy judging; wrong-answer generation and filtering	Apache-2.0
`allenai/wildguard` (Han et al., 2024)	Response-refusal judging	Apache-2.0
`cais/HarmBench-Mistral-7b-val-cls` (Mazeika et al., 2024)	Harmful-response judging	MIT

Table 14: **Datasets and benchmarks with their licenses or terms.**

Asset	License / terms
OR-Bench (Cui et al., 2025)	CC-BY-4.0
XSTest (Röttger et al., 2024)	CC-BY-4.0
HarmBench (Mazeika et al., 2024)	MIT
AdvBench (Zou et al., 2023b)	MIT
sycophancy-eval (Sharma et al., 2024)	MIT (dataset card); source-data terms also apply
TriviaQA (Joshi et al., 2017)	Apache-2.0
MMLU (Hendrycks et al., 2021)	MIT
TruthfulQA (Lin et al., 2022)	Apache-2.0
tinyBenchmarks (Maia Polo et al., 2024)	MIT

This proves Eq. (20). A positive lower bound suffices for local suppression; reducing adapter NLL alone does not establish this alignment for the extracted single-layer shift.

F.4 Transfer across inputs and token positions

LoRA extraction reads one position per input, whereas deployment applies its mean reverse shift at every intervention site. For $z \in \mathcal{E}^-$, let $\mathcal{K}_z$ denote these sites and let $\boldsymbol{g}_{z,k}^{(\ell)}$ be the gradient of its negative masked NLL with respect to an independent additive offset at site k, evaluated with all offsets zero. Assume differentiability in these offsets. Then Eq. (14) gives $\boldsymbol{b}^{(\ell)} = |\mathcal{E}^-|^{-1} \sum_z \sum_{k\in\mathcal{K}_z} \boldsymbol{g}_{z,k}^{(\ell)}$, including downstream effects. Let $\boldsymbol{d}_{z,k}^{(\ell)}$ be the adapted-minus-base block-output shift at that site, using identical fixed inputs and masks without steering. Define

$$
\begin{aligned}
m_{\mathrm{loc}}^{(\ell)} &= \frac{1}{|\mathcal{E}^-|} \sum_z \sum_{k\in\mathcal{K}_z} \langle \boldsymbol{g}_{z,k}^{(\ell)}, \boldsymbol{d}_{z,k}^{(\ell)} \rangle, \\
G_\ell^2 &= \frac{1}{|\mathcal{E}^-|} \sum_z \sum_{k\in\mathcal{K}_z} \|\boldsymbol{g}_{z,k}^{(\ell)}\|_2^2, \qquad E_\ell^2 = \frac{1}{|\mathcal{E}^-|} \sum_z \sum_{k\in\mathcal{K}_z} \|\boldsymbol{d}_{z,k}^{(\ell)} - \bar{\boldsymbol{d}}^{(\ell)}\|_2^2,
\end{aligned}
\tag{28}
$$

where $G_\ell, E_\ell \geq 0$. Here $m_{\mathrm{loc}}^{(\ell)}$ is the margin for hypothetical position-specific reverse shifts, and E_ℓ measures their mismatch with the extracted mean. Cauchy–Schwarz gives

$$
|m^{(\ell)} - m_{\mathrm{loc}}^{(\ell)}| \leq G_\ell E_\ell, \qquad m^{(\ell)} \geq m_{\mathrm{loc}}^{(\ell)} - G_\ell E_\ell. \tag{29}
$$

Thus $m_{\mathrm{loc}}^{(\ell)} > G_\ell E_\ell$, together with the smoothness assumptions of Proposition 1, suffices for local suppression. Neither favorable local alignment nor small transfer error follows from adapter fitting alone.

G Use of Existing Assets

Tables 13 and 14 list the model checkpoints and dataset resources used in our experiments, together with their roles and published licenses or terms.