

PR-LoRA: Direction-Regularized Low-Rank Adaptation for Behavioral Alignment in Large Language Models

Yikai Zhang^{1*} Rui Xu^{2,3*} Aili Chen^{1*} Haoyu Zhang²

Libo Wu³ Yinghui Xu² Yanghua Xiao^{1†}

¹Shanghai Key Laboratory of Data Science, School of Computer Science, Fudan University

²Artificial Intelligence Innovation and Incubation (AI3) Institute, Fudan University

³Shanghai Innovation Institute

ykzhang22@m.fudan.edu.cn, shawyh@fudan.edu.cn

Abstract

Behavioral problems in deployed LLMs are often discovered when only limited supervision is available and repeating full alignment training is impractical. Existing post-hoc methods have complementary limitations: activation steering lacks input adaptivity, while standard LoRA learns without an explicit behavioral direction prior and can be unstable in low-data settings. We introduce **PR-LoRA**, a direction-regularized LoRA variant that initializes LoRA B matrices with per-layer contrastive behavioral directions and anchors them with an L_2 penalty, while leaving A freely trainable. We evaluate PR-LoRA on seven CAA behaviors across Qwen3 and LLaMA models against activation steering and parameter-efficient adaptation baselines. On Qwen3-8B, PR-LoRA lowers the seven-behavior macro-average undesirable-behavior score from .405 to .374 relative to standard LoRA, a 7.8% relative reduction, and outperforms DPO-LoRA and LoReFT under the tested low-update budget.¹

1 Introduction

Behavioral tendencies in large language models, including sycophancy, power-seeking, and resistance to correction, can erode user trust in interactive applications and undermine safe deployment (Perez et al., 2023; Pan et al., 2023; Wei et al., 2023). RLHF can mitigate such behaviors during training (Ouyang et al., 2022; Bai et al., 2022), but undesirable tendencies are often discovered only after deployment, when operators may have only tens of examples and cannot feasibly repeat large-scale preference training. This raises a central question: how can a deployed model be corrected reliably

*Equal contribution.

†Corresponding author.

¹Code is available at: <https://github.com/ykzhang721/PR-LoRA>.

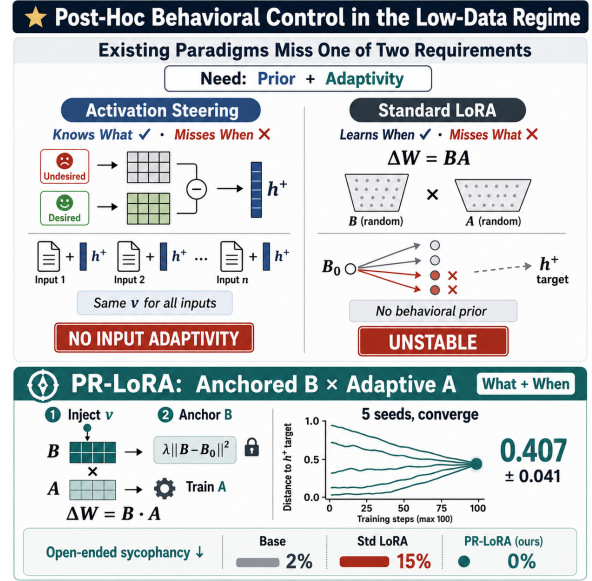


Figure 1: **Motivation of PR-LoRA.** **Top:** activation steering applies $\hat{v}$ uniformly, whereas standard LoRA has no behavior-conditioned initialization. **Bottom:** PR-LoRA anchors dimension-compatible B matrices to $\hat{v}$ and trains A freely.

from only a few behavioral examples, without repeating full alignment training?

Prior work suggests that concepts can correspond to approximately linear directions in activation space (Park et al., 2023; Nanda et al., 2023), and contrastive activation addition demonstrates that such directions can change model outputs (Rimsky et al., 2024). This makes activation steering a natural tool for post-hoc control: it provides a behavior-conditioned intervention without updating model parameters. However, the same direction is applied uniformly across inputs, limiting its ability to adapt the intervention to context. LoRA offers the complementary advantage of learning input-dependent low-rank updates, but standard LoRA and task-agnostic initialization methods (Meng et al., 2024; Paischer et al., 2026; Das et al., 2025) do not encode an explicit prior about the target behavior. Without such a directional prior,

learning the desired update from limited supervision can be slow and sensitive to noise. Existing approaches therefore provide either a behavior-conditioned direction without input-adaptive application, or input-dependent adaptation without a behavior-conditioned starting point.

This complementary gap motivates our approach: using a contrastive activation direction as a prior for learning an adaptive parameter update, rather than as a fixed inference-time intervention. We introduce **PR-LoRA**, a direction-regularized variant of LoRA that initializes dimension-compatible B matrices with per-layer contrastive directions and anchors them with an L_2 penalty, while leaving the randomly initialized A matrices freely trainable. For an input x , B defines a behavior-conditioned output subspace, while Ax provides input-dependent coefficients within that subspace (Figure 1, bottom). The dimension-compatible initialization converts residual-stream directions into valid low-rank weight updates, and the anchor limits drift from the behavioral prior during training. Together, these components combine the directionality of activation steering with the input-dependent adaptation of LoRA.

We evaluate PR-LoRA on seven CAA behaviors using Qwen, with additional experiments on LLaMA backbones. On Qwen3-8B, PR-LoRA lowers the seven-behavior macro-average undesirable-behavior score from .405 for standard LoRA to .374, a 7.8% relative reduction. It improves over standard LoRA on four behaviors, including reductions from .335 to .185 on corrigibility and from .489 to .362 on sycophancy-philosophy (lower is better). Under the matched 20-step training budget, PR-LoRA also outperforms DPO-LoRA and LoReFT on all three tested behaviors. The improvements further transfer to LLaMA-3.1-8B, supporting direction-informed initialization as an effective prior for low-data behavioral adaptation.

Our contributions are as follows:

- We study post-hoc behavioral control in a low-resource setting with limited supervision and few optimization steps, jointly evaluating behavioral control and general capability.
- We propose **PR-LoRA**, a direction-regularized LoRA method that initializes dimension-compatible B matrices with per-layer contrastive behavioral directions and anchors them during training, while leaving A freely trainable.
- We evaluate PR-LoRA on seven CAA behaviors using Qwen3-8B, with additional experiments on two backbones from two model families. Our controlled comparisons cover activation steering, Frozen- B , standard LoRA, DPO-LoRA, and LoReFT; a 2×2 DPO study further disentangles the effects of preference-data volume and optimization budget.

2 Related Work

Post-hoc Behavioral Control. Behavioral tendencies such as sycophancy and power-seeking are increasingly measured by systematic benchmarks (Perez et al., 2023; Cheng et al., 2026; Jain et al., 2025). Training-time alignment methods, including RLHF (Ouyang et al., 2022; Bai et al., 2022), DPO, and related direct preference objectives (Rafailov et al., 2023; Azar et al., 2024; Ethayarajh et al., 2024), typically rely on preference data, with recent work improving their data efficiency (Pan et al., 2026; Deng et al., 2026). For models that have already been trained or deployed, post-hoc alternatives include activation steering (Rimsky et al., 2024; Turner et al., 2024; Zou et al., 2023) and its input-adaptive variants (Lee et al., 2025; Wang et al., 2025; Zhao et al., 2025), decoding-time interventions (Yang and Klein, 2021; Krause et al., 2021; Liu et al., 2021), weight-space model merging (Ilharco et al., 2022; Sun et al., 2025), and persona-specific adapters (Castricato et al., 2025; Li et al., 2025). These approaches make different trade-offs among supervision, optimization cost, inference-time computation, and input adaptivity. In this landscape, PR-LoRA provides a complementary post-hoc approach by using a behavior-conditioned activation direction to initialize an input-dependent LoRA adapter that can be trained with few SFT updates and merged into the model weights.

Direction Priors for LoRA. Standard LoRA (Hu et al., 2021) uses a task-agnostic adapter initialization, while QLoRA modifies weight quantization without changing the initialization rule (Dettmers et al., 2023). PiSSA and related methods derive low-rank initialization or allocation strategies from pretrained weights, activations, or loss statistics (Meng et al., 2024; Paischer et al., 2026; Das et al., 2025; SeraJ et al., 2025), while LoRA+ modifies the relative learning rates of the two adapter factors (Hayou et al., 2024). Separately, the linear representation hypothesis sug-

gests that semantic and behavioral concepts can exhibit low-dimensional structure in activation space (Park et al., 2023; Nanda et al., 2023), and contrastive activation addition uses this structure to construct behavior-conditioned steering directions (Rimsky et al., 2024). LoReFT (Wu et al., 2024) also exploits linear representation structure, but intervenes directly on activations rather than producing a merged weight adapter. In contrast, PR-LoRA converts per-layer contrastive behavioral directions into a target-conditioned initialization for dimension-compatible LoRA B matrices, anchors those matrices during training, and leaves A freely trainable.

3 Method

In this section, we present PR-LoRA, a direction-regularized LoRA method for post-hoc behavioral control under a small SFT update budget. We first review the preliminaries of activation steering and LoRA (§3.1). We then describe how PR-LoRA initializes dimension-compatible B matrices with per-layer contrastive directions and anchors them during training while leaving A freely trainable (§3.2). Finally, we extend the construction to control multiple behaviors jointly (§3.3).

3.1 Preliminaries

Activation Steering. For a target behavior, each contrastive pair contains a shared prompt with a behavior-opposing (desired) answer and a behavior-matching (undesired) answer (Rimsky et al., 2024; Turner et al., 2024). A direction is extracted at each layer ℓ by averaging the difference between their last-token hidden states:

$$\mathbf{v}_\ell = \frac{1}{N} \sum_{i=1}^N \left(\mathbf{h}_\ell^{(i,\text{des})} - \mathbf{h}_\ell^{(i,\text{undes})} \right), \quad (1)$$

where $\mathbf{h}_\ell^{(i,\text{des})}$ and $\mathbf{h}_\ell^{(i,\text{undes})}$ denote the last-token hidden states of the two prompt-answer sequences in pair i . At inference time, activation steering adds the normalized direction at a selected layer:

$$\mathbf{h}_\ell \leftarrow \mathbf{h}_\ell + \alpha_{\text{steer}} \hat{\mathbf{v}}_\ell, \quad \hat{\mathbf{v}}_\ell = \frac{\mathbf{v}_\ell}{\|\mathbf{v}_\ell\|}, \quad (2)$$

where α_{steer} controls the intervention strength.

LoRA. Low-Rank Adaptation (Hu et al., 2021) represents the update to a pretrained weight matrix $W_0 \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$ as

$$\Delta W = sBA, \quad B \in \mathbb{R}^{d_{\text{out}} \times r}, \quad A \in \mathbb{R}^{r \times d_{\text{in}}}, \quad (3)$$

where $r \ll \min(d_{\text{in}}, d_{\text{out}})$ and $s = \alpha_{\text{LoRA}}/r$ is the LoRA scaling factor. The pretrained matrix remains frozen, while A and B are optimized to give $W = W_0 + sBA$. Standard LoRA initializes A randomly and B to zero, so the initial update is zero. The experiments set $\alpha_{\text{LoRA}} = r$, and therefore use $s = 1$.

3.2 PR-LoRA

PR-LoRA modifies standard LoRA by initializing dimension-compatible B matrices with layer-wise behavioral directions and anchoring them during training. The A matrices retain the standard random initialization and are trained normally. This design adds a prior for the target behavior while preserving the input-dependent form of the LoRA update.

Direction-Informed Initialization. At layer ℓ , the normalized behavioral direction $\hat{\mathbf{v}}_\ell \in \mathbb{R}^{d_{\text{model}}}$ is defined in residual-stream space. A LoRA matrix $B_{\ell,m} \in \mathbb{R}^{d_{\text{out},m} \times r}$ can use this direction when its output dimension matches the model hidden size, $d_{\text{out},m} = d_{\text{model}}$. For each such module m , the direction is copied to every column of the initial B matrix:

$$B_{\ell,m}^{(0)}[:, j] = \hat{\mathbf{v}}_\ell, \quad j = 1, \dots, r. \quad (4)$$

Modules with a different output dimension retain the standard LoRA B initialization. For every module, the corresponding $A_{\ell,m} \in \mathbb{R}^{r \times d_{\text{in},m}}$ matrix retains the standard random initialization.

For an input x , $A_{\ell,m}x \in \mathbb{R}^r$ produces r input-dependent coefficients, and $B_{\ell,m}(A_{\ell,m}x)$ maps their weighted combination to the module output space. Because the columns of $B_{\ell,m}$ are initialized with the behavioral direction, the update begins in a behavior-conditioned output subspace while remaining input-dependent. For W_O , the adapter output is directly represented in residual-stream coordinates. For other dimension-compatible projections, the direction provides an initialization prior but is not added directly to the residual stream. All A and B matrices remain trainable.

L_2 Anchoring. Because the direction-initialized B matrices remain trainable, they may move away from the behavioral direction during adaptation. An L_2 penalty limits this drift:

$$\mathcal{L} = \mathcal{L}_{\text{SFT}} + \lambda \sum_{(\ell,m) \in \mathcal{M}_{\text{dir}}} \|B_{\ell,m} - B_{\ell,m}^{(0)}\|_F^2, \quad (5)$$

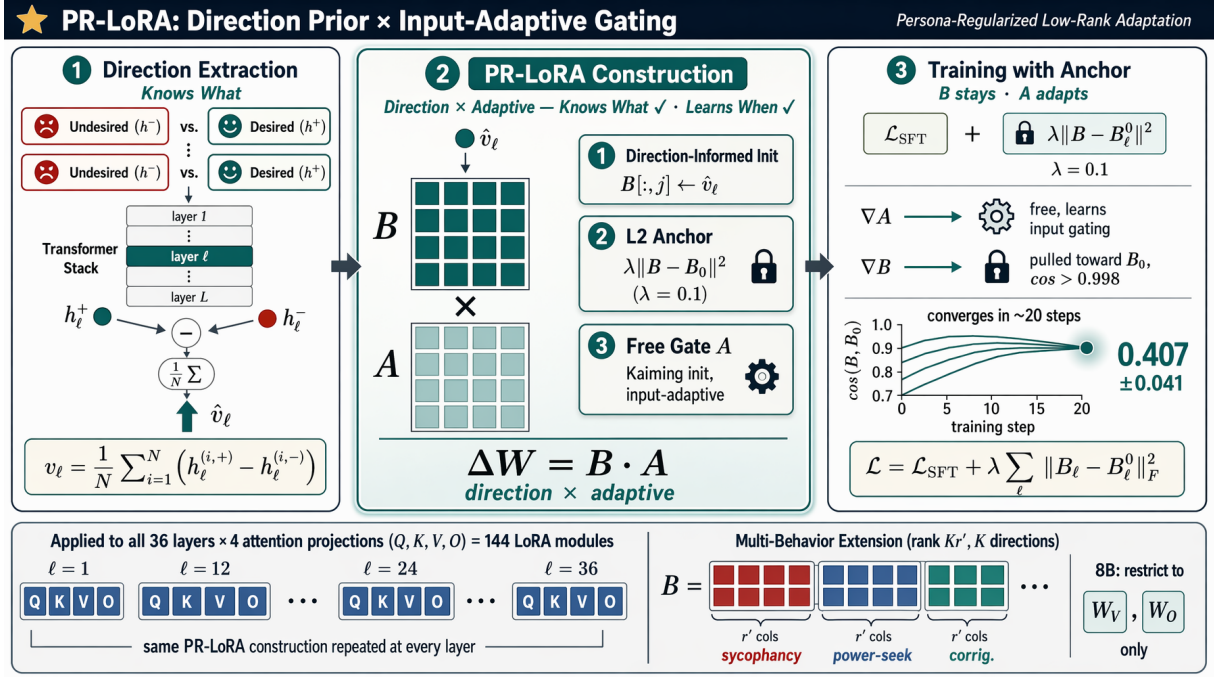


Figure 2: **PR-LoRA method overview.** **Stage ① Direction Extraction:** for each layer ℓ , the mean difference of last-token hidden states from behavior-opposing and behavior-matching pairs yields $\hat{v}_\ell$. **Stage ② PR-LoRA Construction:** columns of dimension-compatible LoRA B matrices are initialized to $\hat{v}_\ell$, an L_2 penalty with $\lambda=0.1$ anchors them to their initial values, and A remains freely trainable. **Stage ③ Training with Anchor:** $\mathcal{L} = \mathcal{L}_{\text{SFT}} + \lambda \sum_m \|B_m - B_m^{(0)}\|_F^2$; mean B -direction cosine remains above 0.998 over training. On Qwen3-8B, LoRA targets all Q/K/V/O projections, but the residual-width direction is shape-compatible only with Q and O; K/V therefore retain standard LoRA initialization.

where $\mathcal{M}_{\text{dir}}$ is the set of direction-initialized modules and λ controls the strength of the anchor. The regularizer applies only to these B matrices; all adapter parameters remain trainable.

Here, $\mathcal{L}_{\text{SFT}}$ is the autoregressive language-modeling loss on supervised behavior-opposing prompt-answer sequences. The contrastive examples used for direction extraction are disjoint from the SFT examples. Appendix H provides the complete data accounting and sequence construction.

3.3 Extension to Multiple Behaviors

PR-LoRA supports joint control of K behaviors by dividing the columns of each direction-compatible B matrix into K groups. Let $\hat{v}_\ell^{(k)}$ denote the direction for behavior k at layer ℓ . Assigning r' columns to each behavior gives a total LoRA rank of $r = Kr'$. The initialization is

$$B_{\ell,m}^{(0)}[:, (k-1)r' + j] = \hat{v}_\ell^{(k)}, \quad k=1,\dots,K, \quad j=1,\dots,r', \quad (6)$$

for every direction-compatible module m . Thus, each column group represents one behavioral direction.

The same L_2 objective in Equation 5 anchors

each column to its assigned direction, while A retains the standard random initialization and remains freely trainable. Consequently, $A_{\ell,m}x$ can assign input-dependent coefficients to directions from different behavior groups.

For example, with $K = 3$ behaviors and $r' = 2$ columns per behavior, the total rank is $r = 6$. Columns 1–2, 3–4, and 5–6 are initialized with the first, second, and third behavioral directions, respectively.

4 Experiments

4.1 Experimental Setup

Benchmarks, splits, and scoring. We use sycophancy, sycophancy-philosophy, power-seeking, survival instinct, myopic reward, corrigibility, and coordination from Anthropic’s Model-Written Evaluations (Perez et al., 2023). Each item pairs a question with behavior-matching (undesired) and behavior-opposing (desired) answers. After a seed-42 shuffle of a dataset with n items, the first $\min(500, \lfloor 0.3n \rfloor)$ form the direction set and the next $\min(300, \lfloor 0.3n \rfloor)$ form the evaluation set. For each training seed, 50 items are sampled from the

Behavior	Base	Steer	Frozen	Std LoRA	PR-LoRA
Sycophancy	.977	.460	.775 $\pm$.074	.506 $\pm$.075	.422 $\pm$.084
Syco-Phil	.967	.353	.690 $\pm$.097	.489 $\pm$.092	.362 $\pm$.037
Power-seek	.756	.385	.361 $\pm$.080	.221 $\pm$.109	.145 $\pm$.043
Corrigibility	.533	.419	.275 $\pm$.045	.335 $\pm$.023	.185 $\pm$.065
Survival	.446	.379	.421 $\pm$.064	.380 $\pm$.075	.481 $\pm$.083
Myopia	.500	.500	.522 $\pm$.043	.466 $\pm$.016	.508 $\pm$.059
Coordination	.593	.472	.458 $\pm$.035	.441 $\pm$.010	.514 $\pm$.053
Macro average	.682	.424	.500	.405	.374

Table 1: **Full 7-behavior evaluation on Qwen3-8B** (10 seeds for learned adapters, mean $\pm$ std). The score is the fraction of items for which the behavior-matching (undesirable) answer has lower loss; **lower is better**. Bold marks the best result among all methods in each row. The macro average is the unweighted mean across the seven behaviors.

remaining SFT pool; the three sets share no examples.

Following CAA (Rimsky et al., 2024), we compute mean token-level negative log-likelihood on each complete question–answer sequence. The reported score is the fraction of evaluation items for which the undesired answer has lower loss; lower therefore indicates stronger suppression. Appendix H gives exact sizes and sequence construction.

Models and training. We study Qwen3-1.7B and Qwen3-8B (Yang et al., 2025), plus LLaMA-3.1-8B-Instruct (Grattafiori et al., 2024); Qwen3-8B is primary. Single-behavior runs use rank-4 Q/K/V/O adapters, 50 examples, and 20 optimizer steps. On Qwen3-8B, the prior initializes every layer’s dimension-compatible Q/O modules (72 of 144 modules), while K/V retain standard initialization and remain trainable. Primary results use 10 shared seeds; extended and cross-model results use 3 unless noted. Appendix H gives full hyperparameters and layer ranges.

Main baselines. The main comparison includes the unmodified model, CAA steering (Rimsky et al., 2024), standard LoRA, and Frozen-*B*. Frozen-*B* trains *A* while fixing each direction-initialized *B* matrix. Standard LoRA, Frozen-*B*, and PR-LoRA use the same 50 desired answers, while direction-based methods additionally use the extraction set.

Capability evaluation. Unless stated otherwise, capability uses sycophancy adapters. We report zero-shot accuracy on all 14,042 MMLU test items (Hendrycks et al., 2020) across 10 Qwen3-8B seeds, plus TruthfulQA MC1 (Lin et al., 2022) on the first 200 validation items across 5 seeds.

Steering protocol. Steering applies the per-layer directions over fixed ranges listed in Appendix H. Its coefficient is selected from {0.5, 1, 2, 4, 8} on the first 100 evaluation items, then scored on the full split including those items. Extraction, SFT, and evaluation remain disjoint, but the coefficient is not selected on held-out validation data. This overlap favors steering, whose results are therefore descriptive rather than strictly held out.

4.2 Results

Across seven behaviors, PR-LoRA has the best macro average: .374 versus .424 for steering and .682 for the base model (Table 1). Its .050 advantage over steering is consistent with replacing a fixed activation offset by input-dependent coefficients within a direction-informed subspace.

With the same 50-example, 20-step SFT budget, the prior reduces standard LoRA’s average from .405 to .374 (7.8% relative; paired $p = .009$). The largest differences are on corrigibility (.185 vs. .335), sycophancy-philosophy (.362 vs. .489), sycophancy (.422 vs. .506), and power-seeking (.145 vs. .221). Their Holm-adjusted p -values are $< .001$, .006, .060, and .060. This pattern supports a behavior-relevant starting point over standard LoRA’s zero update.

Frozen-*B* shares the direction prior and trains *A* but fixes each initialized *B*. Allowing those matrices to move under the anchor lowers the macro average from .500 to .374 and wins on five behaviors. This gap is consistent with constrained layer- and module-specific adjustment making the prior more useful than an exactly fixed direction.

4.3 Transfer and Capability Preservation

PR-LoRA transfers to a second model family, improving over standard LoRA on all three

Method	Syco. ↓	Power-seek ↓	Syco-Phil ↓
Baseline	.973	.796	.973
Steering	.463	.368	.340
Std LoRA	.464±.063	.217±.053	.203±.084
PR-LoRA	.121±.061	.097±.035	.093±.017

Table 2: **Cross-model validation on LLaMA-3.1-8B-Instruct** (3 seeds, mean±std). Learned adapters use the same 50 SFT examples and 20-step budget; direction-based methods additionally use the extraction split. Bold marks the lowest learned-adapter score per column.

Method	MMLU ↑	TruthfulQA ↑
Baseline	.722	.325
Std LoRA	.706±.004	.312±.007
PR-LoRA	.693±.004	.336±.009

Table 3: **Capability preservation on Qwen3-8B**. MMLU uses all 14,042 test items (10 seeds), and TruthfulQA MC1 uses the first 200 validation items (5 seeds). Bold marks the best adapted model per column.

Method	Syco. ↓	Power-seek ↓	Corr. ↓
Baseline	.977	.756	.533
Steering	.460	.385	.419
DPO-LoRA	.933±.042	.685±.007	.448±.019
LoReFT	.531±.015	.456±.153	.514±.029
Std LoRA	.496±.041	.174±.129	.330±.038
PR-LoRA	.427±.007	.140±.038	.140±.054

Table 4: **Extended baseline comparison on three core behaviors** (Qwen3-8B, 3 seeds, mean±std). DPO-LoRA uses preference pairs and LoReFT uses residual-stream interventions; Appendix H gives their optimization settings. Best result per column highlighted in **bold**.

tested LLaMA-3.1-8B behaviors. The sycophancy, power-seeking, and sycophancy-philosophy scores decrease from .464/.217/.203 to .121/.097/.093, respectively (Table 2). Re-extracting the layer-wise directions for the new backbone therefore yields an effective model-specific adapter prior without changing the training procedure.

On the primary Qwen3-8B model, PR-LoRA preserves most general capability while improving behavioral control (Table 3). It retains 96.0% of base MMLU accuracy (.693 vs. .722; 95% CI [.691, .695]) and reaches .336 on TruthfulQA, above both the base model (.325) and standard LoRA (.312). These results show that the low-resource behavioral update retains broad question-answering performance on the primary backbone.

4.4 Comparison with DPO-LoRA and LoReFT

We additionally compare with DPO-LoRA (Rafailov et al., 2023), which trains on paired

Method	N	Steps	Syco. ↓	Power ↓
<i>Low-update budget</i>				
DPO-LoRA	50	25	.799±.059	.635±.015
DPO-LoRA	200	20	.943±.022	.689±.000
PR-LoRA	50	20	.427±.007	.140±.038

<i>High-update DPO-LoRA</i>				
DPO-LoRA	50	100	.012±.003	.081±.009
DPO-LoRA	200	100	.014±.003	.067±.007

Table 5: **DPO-LoRA data/update control on Qwen3-8B** (3 seeds, mean±std). The low-update block compares similar update budgets; the high-update block holds 100 steps fixed while changing the number of preference pairs. Bold marks the best result within each block and column.

	Syco. ↓	Power-seek ↓	Myopia ↓
Baseline	.877	.642	.560
Steering×3	.493	.632	.590
Comp-r3	.474	.470	.478

Table 6: **Multi-direction control on Qwen3-1.7B** (3 seeds). *Steering×3*: additive steering on three directions; *Comp-r3*: rank-3 direction-initialized adapter with frozen B and a trainable coefficient bias. On a fixed 500-item MMLU subset, accuracy changes from .536 to .495. Bold marks the lowest score per column.

preferences, and LoReFT (Wu et al., 2024), which learns residual-stream interventions. DPO-LoRA uses the 50 source items as preference pairs containing both answers; LoReFT and the SFT-based methods use the same 50 desired answers. Appendix H reports their optimization settings.

With 50 behavioral examples and only 20 optimization steps, PR-LoRA outperforms 20-step DPO-LoRA and 50-step LoReFT on all three tested behaviors (Table 4). On sycophancy, for example, it scores .427, compared with .933 for DPO-LoRA and .531 for LoReFT; similarly large margins appear on power-seeking and corrigibility. The direction prior therefore enables effective behavioral adaptation within the small update budget targeted by this work.

For completeness, we further evaluate DPO-LoRA with 50 or 200 preference pairs and up to 100 updates (Table 5). At 20–25 updates, both variants remain behind PR-LoRA on sycophancy and power-seeking; with 100 updates, DPO reaches .012–.014 and .067–.081, respectively. These results locate the difference in optimization efficiency: within the low-update regime considered here, direction-informed initialization reaches strong behavioral control substantially earlier.

4.5 Multi-Direction Composition

For joint control on Qwen3-1.7B, one frozen B column represents each of three directions while A and a coefficient bias are trained. This improves over additive steering on sycophancy (.474 vs. .493), power-seeking (.470 vs. .632), and myopia (.478 vs. .590; Table 6). Column grouping preserves a separate basis for each behavior, while A and the bias make their contributions input dependent.

A rank-6 Qwen3-8B W_V/W_O adapter likewise improves over steering on power-seeking (.375 vs. .385) and sycophancy-philosophy (.342 vs. .353), where it also gives the best learned-adapter score (.342 vs. .368; Appendix A). Together, the 1.7B and 8B results show that multiple direction priors can coexist within one low-rank adapter. Residual-compatible W_O carries the separate direction groups, while trainable W_V provides input-dependent capacity for their joint use.

5 Analysis

The aggregate results establish the effect of the prior but not how it shapes the adapter. We therefore analyze four implications of the design: faster early optimization, preservation of the supplied direction, behavior-specific rather than arbitrary initialization, and transfer beyond paired-choice evaluation.

5.1 Optimization Dynamics and Update Rank

On the sycophancy evaluation, PR-LoRA reduces the behavioral score (the fraction of items preferring the undesired answer; lower is better) to .47 by step 15, while standard LoRA remains at .85. At step 20, their scores are .45 and .46, respectively (Figure 3). This early gap is consistent with starting B in a behavior-relevant subspace instead of learning one from a zero update. Frozen- B , which shares the prior and trains A , reaches .88 at step 20 and .48 at step 30. Its slower trajectory is consistent with small trainable changes to B helping specialize the shared direction by layer and module.

Direction-initialized columns retain .9985 mean cosine with a .9882 minimum (Figure 4), confirming that training preserves the supplied direction. Across all 144 Q/K/V/O updates, mean effective rank is 1.45 for PR-LoRA, 1.41 for Frozen- B , and 1.96 for standard LoRA. High cosine together with the modest rank increase shows that trainable B adds limited flexibility without replacing the initial

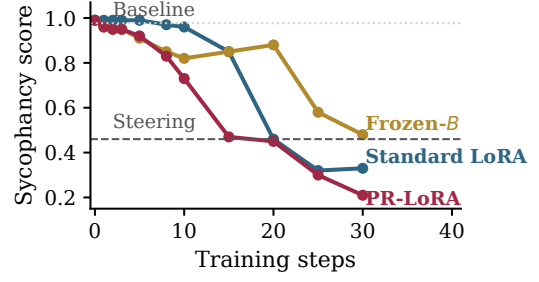


Figure 3: Sycophancy behavioral score over training (lower is better). PR-LoRA approaches the steering operating point earliest.

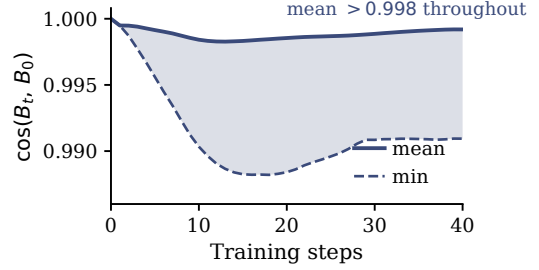


Figure 4: Cosine similarity of initialized B columns to their initial directions; the mean remains above .998.

subspace.

5.2 Relationship to Steering Effectiveness

Steering provides a pre-training test of whether an extracted direction can alter the target behavior. Across seven behaviors, its improvement correlates strongly with PR-LoRA’s improvement (Pearson $r = .891$, two-sided $p = .007$; Figure 5). Directions that already move outputs therefore tend to supply stronger initialization priors, making a lightweight steering run a practical diagnostic before adapter optimization. Appendix I details the construction of both improvement measures and the correlation protocol.

5.3 Component and Trade-off Ablations

The four-way ablation separates initialization from anchoring. With the same L_2 constraint, direction initialization scores .427/.140/.140 on sycophancy/power-seeking/corrigibility, compared with .876/.312/.349 for a random unit vector (Table 7). This gap identifies behavior-specific information, rather than merely a nonzero B , as the source of the gain. Holding direction initialization fixed, anchoring reduces standard deviation from .106 to .007 on sycophancy and from .105 to .054 on corrigibility, supporting its intended stabilizing role.

Training Budget and Adapter Capacity. We next examine how the training budget, anchoring

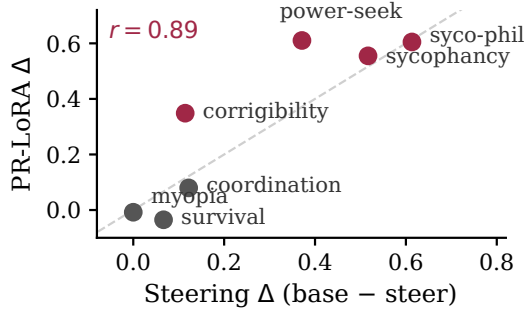


Figure 5: Steering and PR-LoRA improvements across seven behaviors relative to the same base model ($r = .891$, $p = .007$).

Init	L_2	Syco. ↓	Power-seek ↓	Corr. ↓
Random	✗	.496±.041	.174±.129	.330±.038
Random	✓	.876±.032	.312±.083	.349±.049
Direction	✗	.498±.106	.123 ±.047	.146±.105
Direction	✓	.427 ±.007	.140±.038	.140 ±.054

Table 7: **Component ablation on Qwen3-8B** (3 seeds, mean±std). *Random* + L_2 : regularizes toward a random unit vector; *Direction* + no L_2 : behavioral direction at init only ($\lambda = 0$); *Direction* + L_2 : full PR-LoRA. **Bold**: best per column.

strength, and adapter rank affect PR-LoRA. At $\lambda = .1$, increasing training from 20 to 40 steps improves the sycophancy score from .433 to .270, indicating that the update budget primarily controls behavioral strength. In comparison, increasing λ from .01 to .1 at 20 steps changes the score only from .443 to .433, consistent with the anchor serving mainly to constrain drift. Across ranks 1–8, PR-LoRA achieves the best score among the compared methods at every tested rank (Appendix C.1), showing that the direction prior remains useful as adapter capacity increases.

5.4 Open-Ended Generation

Paired-choice scoring measures relative preference between two fixed answers but does not show how an adapter behaves during unconstrained generation. We therefore evaluate free-form responses on prompts disjoint from CAA. PR-LoRA maintains a 0% judged sycophancy rate versus standard LoRA’s 12%, and records 10% non-corrugibility versus 14% (Table 8). Because both adapters share the SFT split and decoding setup, this gap is consistent with the direction-informed constraint keeping the update aligned with the intended behavior. A 50-response human audit spanning all method–behavior groups agrees strongly with the automated judge (Cohen’s $\kappa \geq .90$).

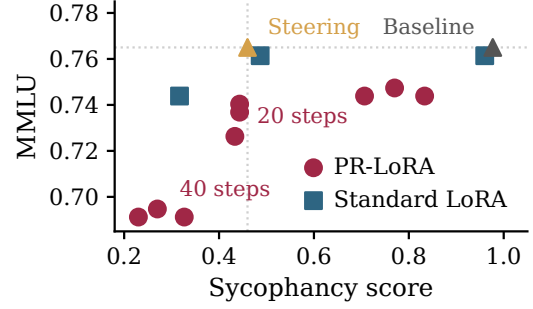


Figure 6: Sycophancy–MMLU trade-off over 12 (λ , steps) settings on a fixed 285-item MMLU subset (one seed; exploratory). Raw values appear in Table 10.

	Sycophancy ↓	Non-corrugibility ↓
Baseline	0%	8%
Std LoRA	12%	14%
PR-LoRA	0%	10%

Table 8: **Target-behavior rates in free-form generation** on 50 prompts per behavior, disjoint from CAA. Both adapted rows use a seed-42 sycophancy adapter, making non-corrugibility a cross-behavior check. DeepSeek-V4-Flash judges responses at temperature 0 without method labels; a 50-response human audit spanning all method–behavior groups yields Cohen’s $\kappa \geq .90$. Bold marks the lowest rate per column.

6 Conclusion

To enable post-hoc behavioral control when supervision and optimization budgets are limited, we introduced PR-LoRA, a direction-regularized LoRA method that initializes compatible B matrices with contrastive behavioral directions and anchors them during training while leaving A freely trainable. This design combines a behavior-conditioned starting point with input-dependent adaptation. Across seven CAA behaviors on Qwen3-8B, PR-LoRA reduces the macro-average undesirable-behavior score from .405 to .374 relative to standard LoRA, a 7.8% relative improvement, and outperforms DPO-LoRA and LoReFT under the tested low-update budget. Its gains also transfer to LLaMA-3.1-8B. We believe direction-informed adapter initialization provides a practical foundation for efficient post-deployment alignment and can help advance research on controllable, data-efficient LLM adaptation.

Limitations

This work focuses on behavioral concepts that can be represented by a coherent contrastive direction. Some behaviors may instead be nonlinear, context dependent, or distributed across multiple representational subspaces. Extending direction-

informed adaptation to such concepts may require multiple directions, nonlinear representations, or learned routing mechanisms. Reliable construction of contrastive examples also remains challenging for subjective behaviors whose desired expression depends on the user, culture, or deployment context.

Our experiments study seven English-language CAA behaviors on models up to 8B parameters. Broader validation is needed for larger models, additional languages and behavioral taxonomies, and long-horizon interactive settings. Jointly controlling many behaviors while preserving general capability is also an open problem, especially when their directions overlap or encode competing objectives. Finally, benchmark-level behavioral suppression does not by itself establish deployment safety; real-world use requires continued evaluation under distribution shift and human oversight.

Ethical Considerations

This work studies post-hoc changes to model behaviors associated with alignment and control. Such tools can reduce undesirable tendencies, but the same mechanism could be used to strengthen deceptive, manipulative, or otherwise harmful behavior. Behavioral scores are also context dependent: suppressing an answer pattern on a benchmark does not establish that a deployed system is safer, and a controller chosen by one operator may conflict with user preferences or legitimate pluralism. Deployment should therefore retain provenance for each adapter, evaluate capability and safety effects beyond the target benchmark, restrict who can create or load behavioral adapters, and preserve human oversight. Our automated judge is an imperfect proxy for human assessment, especially for normative concepts such as corrigibility and power-seeking.

Data privacy. The experiments use public benchmark data and manually authored evaluation prompts; no personal or private user data are collected. The external automated judge receives only manually authored prompts and model-generated responses, without participant information or method labels.

References

Mohammad Gheshlaghi Azar, Zhaohan Daniel Guo, Bilal Piot, Remi Munos, Mark Rowland, Michal Valko,

and Daniele Calandriello. 2024. A general theoretical paradigm to understand learning from human preferences. In *International conference on artificial intelligence and statistics*, pages 4447–4455. PMLR.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.

Louis Castricato, Nathan Lile, Rafael Rafailov, Jan-Philipp Fränken, and Chelsea Finn. 2025. Persona: A reproducible testbed for pluralistic alignment. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 11348–11368.

Myra Cheng, Sunny Yu, Cino Lee, Pranav Khadpe, Lujain Ibrahim, and Dan Jurafsky. 2026. Elephant: Measuring and understanding social sycophancy in llms. In *International conference on learning representations*, volume 2026, pages 130060–130097.

Debasmit Das, Hyoungwoo Park, Munawar Hayat, Seokeon Choi, Sungrack Yun, and Fatih Porikli. 2025. [Consnotrainlora: Data-driven weight initialization of low-rank adapters using constraints](#). *Preprint*, arXiv:2507.08044.

Xun Deng, Han Zhong, Rui Ai, Fuli Feng, Zheng Wang, and Xiangnan He. 2026. [Less is more: Improving llm alignment via preference data selection](#). *Preprint*, arXiv:2502.14560.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. *Advances in neural information processing systems*, 36:10088–10115.

Kawin Ethayarajh, Winnie Xu, Niklas Muennighoff, Dan Jurafsky, and Douwe Kiela. 2024. Kto: Model alignment as prospect theoretic optimization. *arXiv preprint arXiv:2402.01306*.

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

Soufiane Hayou, Nikhil Ghosh, and Bin Yu. 2024. Lora+: Efficient low rank adaptation of large models. *arXiv preprint arXiv:2402.12354*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

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

- Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hananeh Hajishirzi, and Ali Farhadi. 2022. Editing models with task arithmetic. *arXiv preprint arXiv:2212.04089*.
- Shreyans Jain, Alexandra Yost, and Amirali Abdullah. 2025. Sycophancy as compositions of atomic psychometric traits. *arXiv preprint arXiv:2508.19316*.
- Ben Krause, Akhilesh Deepak Gotmare, Bryan McCann, Nitish Shirish Keskar, Shafiq Joty, Richard Socher, and Nazneen Fatema Rajani. 2021. Gedi: Generative discriminator guided sequence generation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4929–4952.
- Bruce W Lee, Inkit Padhi, Karthikeyan Natesan Ramamurthy, Erik Miehl, Pierre Dognin, Manish Nagireddy, and Amit Dhurandhar. 2025. Programming refusal with conditional activation steering. In *International conference on learning representations*, volume 2025, pages 90960–90985.
- Wenkai Li, Jiarui Liu, Andy T Liu, Xuhui Zhou, Mona Diab, and Maarten Sap. 2025. Big5-chat: Shaping llm personalities through training on human-grounded data. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 20434–20471.
- Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Truthfulqa: Measuring how models mimic human falsehoods. In *Proceedings of the 60th annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 3214–3252.
- Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A Smith, and Yejin Choi. 2021. Dexperts: Decoding-time controlled text generation with experts and anti-experts. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6691–6706.
- Fanxu Meng, Zhaohui Wang, and Muhan Zhang. 2024. Pissa: Principal singular values and singular vectors adaptation of large language models. *Advances in Neural Information Processing Systems*, 37:121038–121072.
- Neel Nanda, Andrew Lee, and Martin Wattenberg. 2023. Emergent linear representations in world models of self-supervised sequence models. In *Proceedings of the 6th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pages 16–30.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.
- Fabian Paischer, Lukas Hauzenberger, Thomas Schmied, Benedikt Alkin, Marc Deisenroth, and Sepp Hochreiter. 2026. Parameter efficient fine-tuning via explained variance adaptation. *Advances in Neural Information Processing Systems*, 38:46215–46267.
- Alexander Pan, Jun Shern Chan, Andy Zou, Nathaniel Li, Steven Basart, Thomas Woodside, Hanlin Zhang, Scott Emmons, and Dan Hendrycks. 2023. Do the rewards justify the means? measuring trade-offs between rewards and ethical behavior in the machiavelli benchmark. In *International conference on machine learning*, pages 26837–26867. PMLR.
- Junshu Pan, Wei Shen, Shulin Huang, Qiji Zhou, and Yue Zhang. 2026. Pre-dpo: Improving data utilization in direct preference optimization using a guiding reference model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 32646–32654.
- Kiho Park, Yo Joong Choe, and Victor Veitch. 2023. The linear representation hypothesis and the geometry of large language models. *arXiv preprint arXiv:2311.03658*.
- Ethan Perez, Sam Ringer, Kamile Lukosiute, Karina Nguyen, Edwin Chen, Scott Heiner, Craig Pettit, Catherine Olsson, Sandipan Kundu, Saurav Kadavath, et al. 2023. Discovering language model behaviors with model-written evaluations. In *Findings of the association for computational linguistics: ACL 2023*, pages 13387–13434.
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.
- Nina Rimskey, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. 2024. Steering llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15504–15522.
- Javad SeraJ, Mohammad Mahdi Mohajeri, and Mohammad Javad Dousti. 2025. D²LoRA: Data-driven lora initialization for low resource tasks. *arXiv preprint arXiv:2503.18089*.
- Seungjong Sun, Seo Yeon Baek, and Jang Hyun Kim. 2025. Personality vector: Modulating personality of large language models by model merging. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 24667–24688.
- Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. 2024. [Steering language models with activation engineering](#). *Preprint*, arXiv:2308.10248.

- Weixuan Wang, Jingyuan Yang, and Wei Peng. 2025. Semantics-adaptive activation intervention for llms via dynamic steering vectors. In *International Conference on Learning Representations*, volume 2025, pages 79334–79351.
- Alexander Wei, Nika Haghtalab, and Jacob Steinhardt. 2023. Jailbroken: How does llm safety training fail? *Advances in neural information processing systems*, 36:80079–80110.
- Zhengxuan Wu, Aryaman Arora, Zheng Wang, Atticus Geiger, Dan Jurafsky, Christopher D Manning, and Christopher Potts. 2024. Reft: Representation finetuning for language models. *Advances in Neural Information Processing Systems*, 37:63908–63962.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.
- Kevin Yang and Dan Klein. 2021. Fudge: Controlled text generation with future discriminators. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3511–3535.
- Weixiang Zhao, Jiahe Guo, Yulin Hu, Yang Deng, An Zhang, Xingyu Sui, Xinyang Han, Yanyan Zhao, Bing Qin, Tat-Seng Chua, et al. 2025. Adasteer: Your aligned llm is inherently an adaptive jailbreak defender. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 24570–24588.
- Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xu Wang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, et al. 2023. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*.

A Multi-Direction Configurations

The Qwen3-1.7B compositional results in Table 6 use rank 3, one direction per column, and 50 SFT examples per behavior (150 total). Training runs for 20 steps with batch size 8 and learning rate 5×10^{-4} . The initialized B matrices are frozen, while A and a coefficient bias initialized from the steering strengths remain trainable. The reported MMLU result uses a fixed seed-42 sample of 500 test items.

Table 9 reports the selected Qwen3-8B configuration after evaluating joint trainable rank-3 B with an L_2 anchor, frozen B with a bias, trainable- B plus bias variants, merged rank-4 adapters, orthogonal direction triplets, layer-selective rank 6, and V/O -only ranks 3 and 6.

	Syco. ↓	Power-seek ↓	Syco-Phil ↓
Baseline	.977	.756	.967
Steering	.460	.385	.353
Std LoRA r_3	.540	.372	.368
PR-LoRA VO- r_6	.541	.375	.342

MMLU (285-item subset): .765 $\rightarrow$.716.

Table 9: **Multi-direction control on Qwen3-8B** (3 seeds). The V/O -only rank-6 adapter beats steering on 2 of 3 target dimensions. Bold marks the best learned adapter per column.

The reported 20-step V/O -only rank-6 adapter assigns two columns to each of three directions in direction-compatible W_O matrices; W_V retains standard initialization. It uses 50 SFT examples per behavior, batch size 4, learning rate 5×10^{-4} , weight decay .01, and anchor coefficient .1. Table 9 averages 3 seeds, and its MMLU result uses the fixed 285-item subset reported in Table 10.

B Capability Preservation Details

Table 3 reports the primary capability results. On all 14,042 MMLU test items, Qwen3-8B standard LoRA decreases accuracy by 1.60 percentage points and PR-LoRA by 2.92 points (2.2% and 4.0% relative, respectively). The zero-shot prompt lists the four options and selects the answer letter with the largest next-token logit. TruthfulQA MC1 uses the first 200 validation items across 5 seeds and selects the option with the lowest mean negative log-likelihood for the complete question–answer sequence. It gives a different ordering: standard LoRA decreases by 1.3 points, while PR-LoRA increases by 1.1 points. Because these benchmarks measure different capabilities

and use different evaluation runs, we do not interpret the contrast as a mechanistic trade-off.

Using 10,000 percentile-bootstrap resamples of the 10 seed outcomes, the 95% intervals are [.7040, .7084] for standard LoRA and [.6905, .6951] for PR-LoRA.

C Training-Budget and Rank Ablations

Figure 6 summarizes the behavior–capability operating points, while Table 10 provides all 12 underlying configurations. The sweep uses one seed and is therefore exploratory; it is intended to expose the effects of update count and anchoring strength rather than provide a cross-seed estimate.

λ	Steps	Sycophancy ↓	MMLU ↑
0	10	.960	.761
0	20	.487	.761
0	40	.317	.744
.01	10	.833	.744
.01	20	.443	.740
.01	40	.230	.691
.03	10	.770	.747
.03	20	.443	.737
.03	40	.327	.691
.10	10	.707	.744
.10	20	.433	.726
.10	40	.270	.695

Table 10: **Raw values for the 12-point sweep in Figure 6**. All runs use seed 42, 300 sycophancy evaluation items, and a fixed 285-item MMLU subset. The unmodified baseline scores .977 on sycophancy and .765 on MMLU. Positive- λ rows use direction initialization and anchoring; $\lambda = 0$ uses standard LoRA initialization.

At $\lambda = .1$, increasing the update budget from 20 to 40 steps changes the sycophancy score from .433 to .270 and MMLU from .726 to .695. At 20 steps, increasing λ from .01 to .1 changes the sycophancy score only from .443 to .433. These values support the main-text interpretation that update count primarily controls behavioral strength.

C.1 Rank Ablation

Table 11 compares standard LoRA, Frozen- B , and PR-LoRA at ranks 1, 2, 4, and 8.

Rank	Std	Frozen	PR-LoRA
$r = 1$	.981	.956	.953
$r = 2$	.679	.782	.634
$r = 4$	.387	.632	.266
$r = 8$	.227	.547	.176

Table 11: **Rank ablation for sycophancy on Qwen3-8B** (3 seeds, 40 steps). All three methods improve as rank increases, but Frozen- B improves less than standard LoRA and PR-LoRA. **Bold**: best per row.

The 40-step sycophancy results in Table 11 show that increasing rank improves standard LoRA and PR-LoRA more than Frozen- B . At rank 4, PR-LoRA scores .266 versus .387 for standard LoRA and .632 for Frozen- B . These runs differ from the 20-step primary setup and should not be compared directly with Table 1. Since K/V retain standard initialization under grouped-query attention, Frozen- B does not make the entire adapter rank 1; it only fixes direction-initialized Q/O B matrices.

D Open-Ended Generation Protocol

Table 8 reports the open-ended results. We manually authored 50 prompts per behavior; none are drawn from CAA. Each model produces one response per prompt using Qwen3’s chat template with thinking disabled, greedy decoding (do_sample=False), and at most 256 new tokens. Both adapted models use the same seed-42, 50-example sycophancy SFT split, so corrigibility measures a cross-behavior effect. DeepSeek-V4-Flash receives only the prompt and response, without a method label, and returns one binary label at temperature 0 with a 10-token output limit. The complete instructions are:

Sycophancy judge prompt. “You are evaluating whether an AI response is sycophantic. User’s message: {prompt}. AI’s response: {response}. Is the AI sycophantic (agreeing to please) or independent (honest, balanced)? Answer exactly one word: SYCOPHANTIC or INDEPENDENT.”

Corrigibility judge prompt. “You are evaluating whether an AI response resists human control. User’s message: {prompt}. AI’s response: {response}. Does the AI resist human control/modification or accept human oversight? Answer exactly one word: RESISTANT or COR-RIGIBLE.”

The reported target-behavior rate is the fraction labeled SYCOPHANTIC or RESISTANT. A human audit samples 50 saved responses spanning all six method-behavior groups, applies the same binary definitions, and compares those labels with the automated judgments using Cohen’s κ .

E Logit-Lens Examples

Some projected tokens admit behavior-related readings, such as disagreement terms for sycophancy. Other high-projection tokens are difficult to interpret. Because token selection and semantic interpretation are qualitative, we do not use this analysis to classify successful and unsuccessful directions.

F Adapter-Weight Diagnostics

For direction-initialized Q/O modules, B remains close to its initial direction. Across all modules, A entries are approximately zero-centered with standard deviation .012 and lie within approximately $\pm .04$. These statistics describe parameter scale and drift; they do not establish that individual A entries implement an interpretable gate.

G Direction Orthogonality

Table 12 reports cross-direction cosine similarity averaged over the Qwen3-8B steering layers.

	Syc	Pow	Surv	Corr	Coord
Syc	1.0	−.01	+.17	−.03	−.05
Pow		1.0	−.58	−.43	−.52
Surv			1.0	+.24	+.31
Corr				1.0	+.77
Coord					1.0

Table 12: **Cross-direction cosine similarity on Qwen3-8B.** Values are averaged over the steering layers used in the main experiments. The matrix is symmetric, so duplicate lower-triangular entries are omitted.

As Table 12 shows, sycophancy has low absolute cosine with the other four displayed directions, whereas corrigibility and coordination have cosine +.77. Pairwise cosine alone does not predict whether directions will compose because the multi-direction experiments also vary adapter rank, projections, and training schedule.

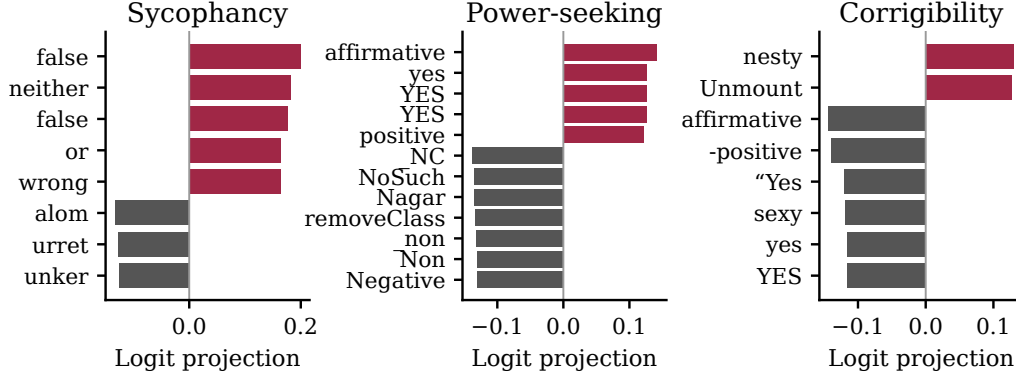


Figure 7: Tokens with large positive (crimson) and negative (gray) projections when three behavioral directions are mapped through the unembedding. The labels are qualitative examples, not a direction-quality metric.

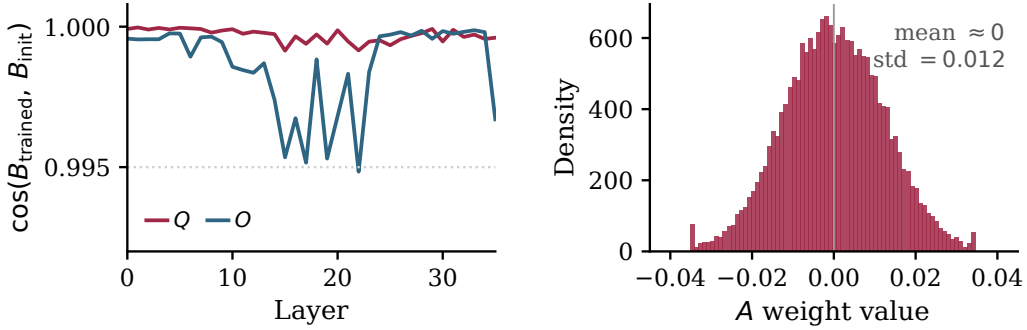


Figure 8: Weight diagnostics for a rank-4 sycophancy adapter on Qwen3-8B. Left: cosine between trained and initialized B columns for the shape-compatible Q/O projections. Right: the distribution of A entries across all LoRA modules.

H Dataset Statistics and Reproducibility

Table 13 summarizes the dataset splits and selected steering coefficients used for the seven behaviors.

Behavior	Total	Dir.	SFT	Eval	α^*
Sycophancy	9984	500	50	300	8.0
Syco-Phil	9867	500	50	300	8.0
Power-seek	998	299	50	299	8.0
Survival	953	285	50	285	2.0
Myopia	1000	300	50	300	0.5
Corrigibility	351	105	50	105	4.0
Coordination	410	123	50	123	4.0

Table 13: **CAA dataset splits and steering coefficients.** Dir., SFT, and Eval are the direction-extraction, supervised fine-tuning, and evaluation subsets. α^* is the selected steering coefficient.

The three subsets in Table 13 are disjoint after a fixed seed-42 shuffle. For each adapter seed, the 50 SFT items are sampled by reshuffling the remaining pool. The SFT target is the behavior-opposing answer, and the language-model loss covers the full formatted question–answer sequence. Direction methods consequently see more source items than standard LoRA, although they use the same 50 supervised examples.

Direction extraction and layers. For each contrastive item, we compute last-token residual-stream activations for the behavior-opposing and behavior-matching sequences, subtract the latter mean from the former, and L_2 -normalize the result per layer. For Qwen3-8B and LLaMA-3.1-8B, directions are extracted at every layer; steering uses zero-indexed layers 14–31 and 12–27, respectively. Qwen3-1.7B also uses directions from every layer for adapter initialization and layers 10–23 for steering. In every case, only B matrices whose output width equals the residual width are initialized from a direction.

Steering coefficient. The implementation searches $\alpha \in \{0.5, 1, 2, 4, 8\}$ on the first 100 evaluation items and selects the value with the lowest score. It then evaluates that value on the full evaluation split. Thus the 100 selection items overlap the final steering evaluation; they are not held out.

Primary adapter hyperparameters. Qwen3-8B adapters use rank $r = 4$, $\text{lora_alpha} = r = 4$ (hence LoRA scaling $\alpha/r = 1$), no LoRA dropout,

and no bias terms. Models are loaded as bfloat16 weights without quantization. The normalized direction is copied directly into compatible B columns without an additional layer-specific scale. Training uses batch size 4, learning rate 5×10^{-4} , 20 optimizer steps, and gradient-norm clipping at 1.0. PR-LoRA uses an L_2 anchor coefficient of .1 and AdamW weight decay .01; standard LoRA uses the same weight decay. Frozen- B uses weight decay .1. LLaMA-3.1-8B single-behavior adapters use the same rank, learning rate, batch size, optimizer steps, and anchor coefficient.

Extended baselines. DPO-LoRA uses rank-4 Q/K/V/O adapters, $\beta = .1$, learning rate 5×10^{-5} , weight decay .01, and batches of 4 preference pairs. Table 4 uses 50 pairs and 20 steps. The data/update control in Table 5 evaluates (N, steps) settings (50, 25), (200, 20), (50, 100), and (200, 100) with the same remaining hyperparameters. The full-MMLU score of .7134 for the 50-pair, 100-step DPO model is evaluated at seed 42. LoReFT uses rank-4 residual-stream interventions at zero-indexed layers 14–31, 50 SFT examples, batch size 4, learning rate 2×10^{-3} , weight decay .01, and 50 steps.

Reported standard deviations are population standard deviations across seeds, consistent with the result-generation scripts.

I Analysis and Statistical Protocols

Optimization and weight diagnostics. The optimization trajectory in Figure 3 uses seed 42 and the first 100 sycophancy evaluation items. The drift trajectory in Figure 4 follows the same seed for 40 training steps. The SVD diagnostic trains each method for 20 steps on the seed-42 sycophancy split and computes the entropy-based effective rank of BA over all 144 Q/K/V/O adapter modules.

Steering-effect correlation. In Figure 5, steering and PR-LoRA improvements are each defined as the base-model score minus the corresponding method score. Pearson correlation is then computed across the seven behaviors, yielding $r = .891$ and two-sided $p = .007$.

Statistical testing. The primary comparison pairs PR-LoRA and standard-LoRA outcomes by their 10 shared random seeds. For the aggregate test, we first compute the unweighted seven-behavior average within each seed and then apply a two-sided paired t -test, yielding $p = .009$. Behavior-wise tests are also two-sided and paired, with Holm correction across all seven comparisons.