SFWE: MITIGATING SPURIOUS FEATURE RELIANCE IN LLM REASONING VIA WEIGHT EDITING

Anonymous authors
Paper under double-blind review

ABSTRACT

Reasoning in large language models can be disrupted by simple perturbations (e.g., adding irrelevant distractors or rephrasing a problem's scenario), suggesting that their reasoning relies on spurious correlations. Identifying and suppressing the spurious features involved in extended chain-of-thought reasoning remains challenging. We introduce Spurious Feature Weight Editing (SFWE), a training-free method that extracts behavioral directions associated with spurious reliance by comparing activations on paired original and perturbed problems, localizes the contributing MLP neurons and attention heads along these directions, and suppresses their influence by removing the direction-aligned components from the selected output weights. Using two small datasets for extracting behavioral vectors and calibration, SFWE improves accuracy on perturbed inputs by 10.9 percentage points on average and accuracy on original inputs by 0.9 points across five open-weight reasoning models and five reasoning benchmarks, reducing the average relative gap between original and perturbed accuracy from 17.4% to 2.2%. Analyses identify two recurring forms of spurious reasoning—over-integrating problem conditions and overthinking—and show that SFWE improves reasoning robustness by suppressing the corresponding behavioral directions.

1 INTRODUCTION

Large language models (LLMs) have made remarkable progress in reasoning tasks (Team et al., 2026; OpenAI, 2025; Guo et al., 2025). However, recent studies show that even strong models can fail under simple perturbations, such as rephrasing a problem's scenario, changing numerical values, or adding irrelevant distractors (Huang et al., 2026; Mirzadeh et al., 2025). This brittleness suggests that their reasoning relies on spurious correlations, making their success sensitive to how a problem is presented rather than solely to the reasoning it requires.

Prior work on reducing reliance on spurious correlations largely targets discriminative tasks, where a model predicts a single label (Mahabadi et al., 2020; Wang et al., 2022; Zhou & Zhu, 2025), and is not designed for long chain-of-thought (CoT) reasoning (Chen et al., 2026). In long CoT, a spurious feature can perturb one intermediate step and propagate through the rest of the trace, making it hard to tell which feature or step caused the final failure. Recent work mitigates spurious reliance in CoT through confounding-token pruning and distillation, or through causal intervention (Guo et al., 2026; Xie et al., 2025; Wu et al., 2024), improving final outcomes but leaving open which intermediate behaviors reflect spurious reliance and which model components implement them.

To bridge this gap, our intuition is that identifying and characterizing spurious features not only provides an interpretable account of reasoning failures, but also enables us to suppress their influence, thereby increasing the relative contribution of causally relevant features and improving reasoning robustness and generalization. Therefore, we introduce Spurious Feature Weight Editing (SFWE), an training-free method that represents and suppress spurious reliance as directions in activation space over the course of CoT generation, building on prior work showing that behaviors such as reasoning patterns and personality traits can be represented in this form (Turner et al., 2023; Rimsky et al., 2024; Chen et al., 2025). As illustrated in Figure 1(a), we treat each answer-preserving perturbation as an input-level intervention that changes a presentation factor while holding the underlying task answer fixed. We retain pairs for which the model succeeds on the original but fails on the perturbed input, and compare their reasoning activations at complementary token and sequence levels

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.

to obtain spurious feature behavioral directions (Figure 1(b)). We use these directions to identify associated attention heads and MLP neurons and perform a one-time orthogonal edit of their output weights Figure 1(c). The resulting model can be applied to subsequent problems without further modification and becomes less sensitive to the targeted spurious features. To the best of our knowledge, SFWE is the first work in long-CoT reasoning to combine mechanistic interpretability with answer-preserving, input-level counterfactual interventions, motivated by Pearl's causal hierarchy (Pearl, 2009), to identify and suppress spurious reliance at both token and sequence levels, thereby improving robustness and generalization under answer-preserving perturbations.

We evaluate SFWE on five open-weight reasoning models across five reasoning benchmarks and their variants augmented with irrelevant distractors. Using up to 20 original–perturbed pairs for extraction and 100 problems for calibration, SFWE improves accuracy on perturbed inputs by 10.9 percentage points and on original inputs by 0.9 points on average, while reducing the relative gap between original and perturbed accuracy from 17.4% to 2.2%. These results demonstrate the effectiveness of SFWE across model families, scales, and benchmarks.

To characterize the behaviors associated with reasoning brittleness and clarify the mechanism underlying SFWE's effectiveness, we analyze the behaviors captured by the extracted behavioral vectors. Across five models, our analysis reveals two recurring behavioral patterns: *over-integration*, in which the model incorporates both solution-relevant conditions and irrelevant distractors into its reasoning without distinguishing between them, and *overthinking*, in which the model repeatedly questions, revises, or abandons otherwise viable reasoning steps. Targeted causal interventions further show that editing the corresponding directions suppresses these behaviors, providing evidence that SFWE improves robustness by reducing these two forms of spurious reasoning.

Our contributions are twofold: 1. We introduce SFWE, an training-free method that identifies spurious feature directions in activation space and uses them to guide targeted weight edits for more robust reasoning. 2. We identify two recurring behaviors associated with spurious reliance—*over-integration* and *overthinking*—and show that our edits improve reasoning robustness by suppressing these behaviors.

We release SFWE at https://anonymous.4open.science/r/SpuriousFeatureWeightEditing-5C29/.

2 PRELIMINARIES

Steering directions. Prior work has shown that certain behavioral attributes of language models can be associated with directions in hidden-state space (Turner et al., 2023; Rimsky et al., 2024; Chen et al., 2025). A common approach estimates a steering vector by subtracting the mean hidden representation associated with an opposing behavior from that associated with the target behavior. For example, contrasting mean activations on harmful requests that models typically refuse with those on harmless requests they answer yields a refusal direction (Arditi et al., 2024), while contrasting "Think Step by Step" and "Answer Immediately" prompts yields a chain-of-thought steering vector (Zhang & Viteri, 2024). For paired examples, let (z_i^+, z_i^-) denote a contrasting pair, and let S_i^+ and S_i^- denote their selected token positions. Writing $h_{\ell,t}(z)$ for the residual-stream activation at layer ℓ and token position t for example z, we obtain a representation of each example by averaging over the selected positions, and estimate a steering vector by averaging the activation differences across the N pairs:

$$\bar{h}^r_{\ell,i} = \frac{1}{|S_i^r|} \sum_{t \in S_i^r} h_{\ell,t}(z_i^r), \quad r \in \{+,-\}, \qquad v_\ell = \frac{1}{N} \sum_{i=1}^{N} \left(\bar{h}^+_{\ell,i} - \bar{h}^-_{\ell,i} \right). \tag{1}$$

The selected positions may consist of a single token or a span, depending on the extraction procedure. Activation steering uses this vector to modify residual-stream activations during inference via $h'_\ell = h_\ell + \alpha v_\ell$, where α controls the sign and magnitude of the intervention. These directions provide targets for behavioral intervention.

Residual-stream decomposition. In pre-norm Transformers (Xiong et al., 2020), attention and MLP sublayers contribute additively to the residual stream (Elhage et al., 2021). For an input context s, let $r_\ell(s)$ denote the residual state at the current token position entering layer ℓ. The layer updates

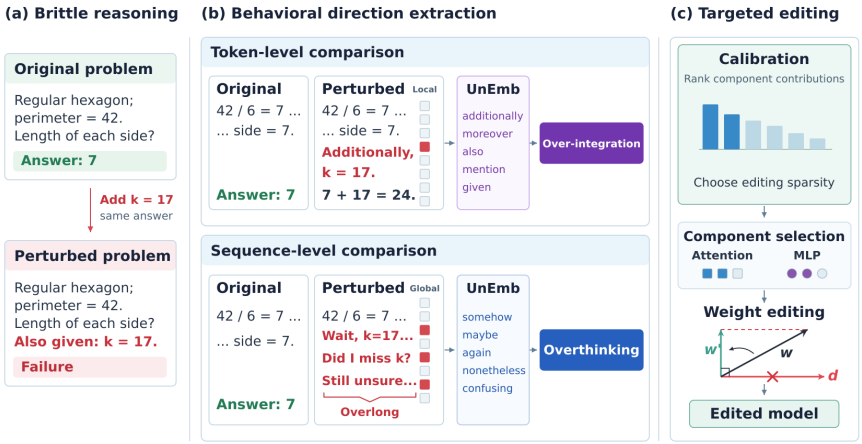

Figure 1: Overview of Spurious Feature Weight Editing (SFWE). (a) Adding an irrelevant condition leaves the correct answer unchanged but can cause the model to fail. SFWE uses pairs in which the original problem is answered correctly and the perturbed problem is answered incorrectly. (b) Token-level extraction compares the same reasoning trace under both prompts at positions most affected by the perturbation, capturing local uptake of irrelevant information (over-integration). Sequence-level extraction compares activations averaged over independently generated traces, capturing broader changes illustrated by repeated doubt and an overlong response (overthinking). Red highlights illustrate these local and distributed behaviors; the UnEmb word lists illustrate how the extracted directions can be interpreted. (c) SFWE ranks attention heads and MLP neurons by their contributions to the directions, uses a separate calibration set to choose how many components to edit, and removes direction-aligned components from their output weights.

this state as

$$r_\ell^{\text{attn}}(s) = r_\ell(s) + \Delta r_\ell^{\text{Attn}}(s), r_{\ell+1}(s) \qquad = r_\ell^{\text{attn}}(s) + \Delta r_\ell^{\text{MLP}}(s), \tag{2}$$

where the attention and MLP updates are computed from the normalized layer input and intermediate residual state, respectively. Each sublayer update can be further decomposed into contributions from individual attention heads or MLP neurons (Elhage et al., 2021; Geva et al., 2021):

$$\Delta r_\ell^{\text{Attn}}(s) = \sum_h W_O^{\ell,h} z_{\ell,h}(s) + b_\ell^{\text{Attn}}, \qquad \Delta r_\ell^{\text{MLP}}(s) = \sum_j a_{\ell,j}(s) W_{\text{down}}^\ell[:, j] + b_\ell^{\text{MLP}}. \tag{3}$$

Here, $z_{\ell,h}(s)$ is the output of attention head h before the output projection, and $W_O^{\ell,h}$ is the corresponding block of columns of the attention output matrix W_O^ℓ. For MLP neuron j, $a_{\ell,j}(s)$ is its scalar activation entering the output projection, including the gating operation in gated MLPs, and $W_{\text{down}}^\ell[:, j]$ is its output weight vector. The bias terms are zero when absent from the architecture. We refer to each attention head or MLP neuron as a *unit*. Its contribution to the residual stream is therefore

$$c_{\ell,h}^{\text{Attn}}(s) = W_O^{\ell,h} z_{\ell,h}(s), \qquad c_{\ell,j}^{\text{MLP}}(s) = a_{\ell,j}(s) W_{\text{down}}^\ell[:, j]. \tag{4}$$

We use $c_u(s) \in \mathbb{R}^{d_{\text{model}}}$ to denote either type of unit contribution.

Directional attribution. Direct logit attribution (DLA) measures a model component's direct contribution to a target token's logit (Elhage et al., 2021). Intuitively, it measures how much the component's residual-stream output contributes along the target token's unembedding direction. For a unit u with output $c_u(s)$ and a target token k, this contribution is

$$\text{DLA}_{u,k}(s) = w_k^\top L_s c_u(s), \tag{5}$$

where w_k is the unembedding vector for token k and L_s is the linear part of the final normalization with its normalization statistics fixed at their observed values. The same projection principle can quantify a unit's contribution along a steering direction. Intuitively, we measure how much of the

unit's residual-stream output lies along the target direction at the same layer. Let d_ℓ^g denote a normalized target direction, indexed by g, in the residual space of layer ℓ. For a unit u in that layer, we define its directional contribution as

$$A_u^g(s) = (d_\ell^g)^\top c_u(s). \tag{6}$$

Since the target direction is defined directly in the same residual space as the unit output, this score requires no final normalization or unembedding. Positive scores indicate that the unit writes along the target direction, while negative scores indicate that it writes against it.

3 Spurious Feature Weight Editing

SFWE identifies behavioral directions associated with reliance on spurious features and uses them to guide targeted output-weight edits. We first describe the construction and selection of original–perturbed pairs (Section 3.1) and the extraction of behavioral vectors (Section 3.2). We then explain component ranking and selection (Section 3.3) and directional weight editing (Section 3.4). Finally, we describe component assignment and calibration of the editing fractions (Section 3.5).

3.1 Original–Perturbed Pairs

We adopt the NoOp perturbation of GSM-Symbolic (Mirzadeh et al., 2025), which causes substantially larger accuracy drops than changing names or numerical values: for each original problem x^{o} we construct x^{p} by adding seemingly relevant but ultimately irrelevant information without changing the correct answer. We generate traces for both with the unedited model and retain pairs whose original is answered correctly and whose perturbed version is not. For each model we sample up to N_{ext} qualifying pairs (all of them if fewer qualify), giving an extraction set $\mathcal{D}_{\mathrm{ext}} = \{(x_i^{\mathrm{o}}, x_i^{\mathrm{p}})\}_{i=1}^N$ kept separate from calibration and evaluation data.

3.2 Steering Direction Extraction

Behavioral changes under a perturbation may extend across a trace or arise at particular positions, so we extract a token-level (TL) and a sequence-level (SL) direction. Let $y_i^{\mathrm{o}}, y_i^{\mathrm{p}}$ be the unedited model's responses. Both directions are estimated as shrunken effect sizes rather than raw means: a coordinate whose value swings widely across pairs is unreliable evidence of a systematic shift, so we divide by its across-pair spread. For per-pair difference vectors $\{\delta_{\ell,i}\}_{i=1}^N$ at layer ℓ, let μ_ℓ and σ_ℓ be their coordinatewise mean and standard deviation over the N pairs, and define

$$\mathcal{V}_\ell\big[\{\delta_{\ell,i}\}_{i=1}^N\big] = \mu_\ell \oslash \big(\sigma_\ell + \lambda \,\mathrm{med}(\sigma_\ell)\big), \tag{7}$$

where $\oslash$ denotes coordinatewise division, $\mathrm{med}(\sigma_\ell)$ is the median entry of σ_ℓ, and $\lambda = 0.25$ shrinks the denominator so that coordinates with near-zero spread are not amplified.

Token-level extraction. We teacher-force each reference trace y_i^r under both prompts and compute the next-token divergence $D_{i,t}^r = D_{\mathrm{KL}}\big(p_\theta(\cdot \mid x_i^{\mathrm{p}}, y_{i,<t}^r) \,\|\, p_\theta(\cdot \mid x_i^{\mathrm{o}}, y_{i,<t}^r)\big)$. We keep as S_i^r the fewest top-ranked positions covering 50% of the total divergence, and average the activation differences taken immediately before predicting $y_{i,t}^r$ over those positions:

$$\delta_{\ell,i}^r = \frac{1}{|S_i^r|} \sum_{t \in S_i^r} \left[h_\ell(x_i^{\mathrm{p}}, y_{i,<t}^r) - h_\ell(x_i^{\mathrm{o}}, y_{i,<t}^r)\right]. \tag{8}$$

The two reference traces select different positions and carry different divergence scales, so each is normalized on its own before the two are pooled:

$$v_\ell^{\mathrm{TL}} = \frac{1}{2} \sum_{r \in \{\mathrm{o},\mathrm{p}\}} \mathcal{V}_\ell\big[\{\delta_{\ell,i}^r\}_{i=1}^N\big]. \tag{9}$$

Fixing the prefix separates prompt-induced changes from differences in generated history, and using both reference traces reduces dependence on either. Each nonzero layerwise vector is then normalized, $d_\ell^g = v_\ell^g / \|v_\ell^g\|_2$ for $g \in \{\mathrm{SL}, \mathrm{TL}\}$.

Sequence-level extraction. Following contrastive CoT steering (Zhang & Viteri, 2024), we mean-pool the residual-stream activation $h_\ell(x_i^r, y_{i,\le t}^r)$ after each of the T_i^r response tokens (excluding the prompt) of each independently generated trace, $r \in \{\mathrm{o}, \mathrm{p}\}$, and apply Equation 7 to the per-pair differences:

$$v_\ell^{\mathrm{SL}} = \mathcal{V}_\ell\big[\{\bar{h}_{\ell,i}^{\mathrm{p}} - \bar{h}_{\ell,i}^{\mathrm{o}}\}_{i=1}^{N}\big], \qquad \bar{h}_{\ell,i}^{r} = \frac{1}{T_i^r}\sum_{t=1}^{T_i^r} h_\ell(x_i^r, y_{i,\le t}^r). \tag{10}$$

This captures broad trajectory differences without requiring token alignment.

3.3 Component Ranking and Selection

For each direction we score every unit with Equation 6, averaging over the token contexts $\mathcal{C}_q^g$ used to score each of M_g attribution traces from the extraction set and then across traces, so each trace has equal weight: $\bar{A}_u^g = \frac{1}{M_g}\sum_{q=1}^{M_g} |\mathcal{C}_q^g|^{-1} \sum_{s\in\mathcal{C}_q^g} A_u^g(s)$. For $g \in \{\mathrm{SL}, \mathrm{TL}\}$ and unit type $m \in \{\mathrm{MLP}, \mathrm{Attn}\}$, let $\mathcal{U}_{g,m}^+$ contain the units of type m, across all layers, with $\bar{A}_u^g > 0$. Each of the four direction–type groups has its own budget: we select the top fraction τ,

$$\mathcal{E}_{g,m}(\tau) = \mathrm{Top}_k\left(\mathcal{U}_{g,m}^+; \bar{A}^g\right), \qquad k = \max\big(1, \lfloor \tau |\mathcal{U}_{g,m}^+| \rfloor\big) \text{ for } \tau > 0. \tag{11}$$

Heads are ranked only against heads and neurons against neurons; pooling them would let τ shift the edit between unit types rather than vary its extent.

3.4 Directional Weight Editing

We adapt weight orthogonalization (Arditi et al., 2024) to the selected units, related to the weight editing of ThinkEdit (Sun et al., 2025). With $W_u = W_{\mathrm{down}}^{\ell}[:, j]$ for neuron j or $W_O^{\ell,h}$ for head h,

$$W_u' = \left(I - d_\ell^g (d_\ell^g)^\top\right) W_u. \tag{12}$$

For any input a_u (a neuron activation or a head's pre-projection output), the edited output $c_u' = W_u' a_u$ satisfies $(d_\ell^g)^\top c_u' = 0$, while its component orthogonal to d_ℓ^g is unchanged. Unselected units are untouched, and the edit is applied once to the weights, requiring neither training nor activation interventions at inference.

3.5 Component Assignment and Sparsity Calibration

On a calibration set $\mathcal{D}_{\mathrm{cal}}$ disjoint from the extraction and test sets, with directions and rankings fixed, we sweep a grid $\mathcal{T}_{g,m}$ of sparsities for each group in isolation, editing only $\mathcal{E}_{g,m}(\tau)$. Each candidate is scored by $\mathcal{S}_{g,m}(\tau) = \frac{1}{2}[\mathrm{Acc_o} + \mathrm{Acc_p}]$ on $\mathcal{D}_{\mathrm{cal}}$, so a sparsity that narrows the gap by degrading original-problem accuracy is not rewarded. Selecting τ per model would fit the evaluation data, so we choose one τ per group by charging each candidate the accuracy it gives up relative to each model's own best:

$$R_{g,m}(\tau) = \frac{1}{|\mathcal{K}|}\sum_{k\in\mathcal{K}}\Big(\max_{\tau'\in\mathcal{T}_{g,m}} \mathcal{S}_{g,m}^{(k)}(\tau') - \mathcal{S}_{g,m}^{(k)}(\tau)\Big), \qquad \tau_{g,m}^* = \arg\min_{\tau\in\mathcal{T}_{g,m}} R_{g,m}(\tau), \tag{13}$$

where $\mathcal{K}$ is the set of models. Regret is zero only for a τ optimal on every model; otherwise it is the cost, in accuracy points, of one global choice.

The final model edits MLP neurons along TL directions and attention heads along SL directions: TL directions capture differences at individual positions, where MLPs transform representations, while SL directions summarize whole trajectories, and attention moves information across positions (Vaswani et al., 2017). Other assignments, including applying both directions to both unit types, are evaluated as ablations (Section 4.2). The configuration $(\tau_{\mathrm{SL,Attn}}^*, \tau_{\mathrm{TL,MLP}}^*)$ is then frozen and used unchanged on all held-out datasets.

4 Experiments

We evaluate whether SFWE improves reasoning accuracy under irrelevant-information perturbations while preserving performance on the original problems. We first describe the experimental

settings (Section 4.1), then compare SFWE with the unedited models, activation steering, and single-granularity editing variants (Section 4.2). We further test the consistency of SFWE's gains across base-model families, model scales (Section 4.3).

4.1 SETTINGS

Models and data. We evaluate five reasoning models: DeepSeek-R1-Distill-Qwen-1.5B, -7B and -32B, DeepSeek-R1-Distill-Llama-8B (Guo et al., 2025), and Qwen3-4B (Yang et al., 2025a). We report per-dataset results for DeepSeek-R1-Distill-Qwen-32B and Qwen3-4B, while for the remaining three models we report accuracy averaged over all five benchmarks under each setting. We use five reasoning benchmarks: Math500, AMC (Mathematical Association of America, n.d.b) (2023–2025, 127 problems), Gaokao-Math (Liao et al., 2024), AIME (Mathematical Association of America, n.d.a) (2024–2026, 90 problems), and GSM-Symbolic (Mirzadeh et al., 2025), each in its original and perturbed (NoOp) form. Following Mirzadeh et al. (2025) and Fu et al. (2026), NoOp variants are generated with GPT-5.4-mini, which adds irrelevant information while preserving the original wording and correct answer.

Baselines and variants. We compare against the unedited model (*Base*) and activation steering (*Steering*), which suppresses the extracted directions in activations at inference (Turner et al., 2023; Rimsky et al., 2024); for Steering, we report the better of its SL and TL variants. We also include ThinkEdit (Sun et al., 2025) as the closest prior weight-editing baseline. We adapt its sequence-level component-editing procedure to our answer-preserving original–perturbed pairs. Weight-editing variants are named by direction and edited unit type (e.g., *TL-MLP*, *SL-Attn*; *-Both* edits heads and neurons). *SL+TL* combines the two directions and is the full SFWE method.

Inference and hyperparameters. We use vLLM (Kwon et al., 2023) in `bfloat16` with greedy decoding and a limit of 16,384 generated tokens, after which generation stops even if the response is incomplete. We set $N_{\text{ext}} = 20$ Math500 pairs per model and calibrate on the first 100 Math500 problems and their NoOp variants, which are excluded from all evaluation. We sweep $\tau \in \{0.25, 1, 2, 5, 10, 25, 100\}\%$ for heads and $\{0.01, 0.05, 0.25, 1, 5, 25, 100\}\%$ for neurons. Minimizing regret (see Table 4 in Appendix A) selects $\tau^* = 25\%$ for TL–MLP and 100% for SL–Attn in the final model, and 5% for SL–MLP and 25% for TL–Attn in the ablations.

4.2 MAIN RESULTS

Table 1 reports per-dataset results for the two models used in the main comparison. Both TL and SL edits improve Average NoOp accuracy for both models: TL reaches 74.8% for DeepSeek-R1-Distill-Qwen-32B and 73.5% for Qwen3-4B, while SL reaches 74.2% and 72.4%, respectively. Combining the two directions achieves the largest Average NoOp gain for both models, raising accuracy from 66.6% to 75.3% ($+8.7$ points) for the 32B model and from 66.0% to 74.1% ($+8.1$ points) for Qwen3-4B. It also obtains the largest per-dataset gains on AMC for the 32B model ($+15.0$ points) and on GSM-Symbolic for Qwen3-4B ($+14.1$ points). Although the directions are extracted from only 20 Math500 pairs, the combined edit improves NoOp accuracy on every dataset for both models. Activation steering with the same directions achieves 72.7% and 63.6% Average NoOp accuracy, suggesting that removing directional contributions from high-contribution units is more effective than applying the directions to hidden activations, consistent with prior work showing that global steering overlooks differences between components (Sun et al., 2026). Orig accuracy remains close to its pre-edit level for most entries in this table, although occasional trade-offs remain; for example, SL+TL on Qwen3-4B lowers AIME Orig accuracy from 55.0% to 38.3%.

SFWE closes the original–perturbed accuracy gap. The unedited models exhibit substantial relative gaps between original and perturbed problems: 12.6% for DeepSeek-R1-Distill-Qwen-32B and 14.1% for Qwen3-4B. TL editing reduces these gaps to 0.8% and 2.5%, respectively, whereas SL editing leaves larger gaps of 4.7% and 7.1%. The combined SL+TL edit yields gaps of 2.0% and 1.2%, achieving the smallest gap for Qwen3-4B and bringing both models close to parity between original and perturbed inputs. These results indicate that token-level contrasts are particularly effective for identifying and suppressing perturbation-sensitive reliance.

Table 1: Main results. Weight edits use sparsities calibrated on the 100 questions of Math500: 25% for TL-MLP, 25% for TL-Attn, 5% for SL-MLP, and 100% for SL-Attn. The TL and SL rows report, for each column independently, the best of that direction's three variants (MLP only, Attn only, both), so neither row corresponds to a single configuration. Gap (%) is computed from the Average columns as $100 \times |\text{Orig} - \text{NoOp}|/\text{Orig}$, serving as a measure of the model's reliance on spurious features during reasoning (Fu et al., 2026). Bold marks the best NoOp or Gap result in each column. See Table 5 in Appendix B for the full per-unit-type breakdown.

Method	GSM-Sym		Math500		AMC		Gaokao-Math		AIME		Average		
	NoOp	Orig	NoOp	Orig	NoOp	Orig	NoOp	Orig	NoOp	Orig	NoOp	Orig	Gap (%)
DeepSeek-R1-Distill-Qwen-32B													
Base	80.1	87.5	79.4	85.9	67.7	77.2	62.3	71.9	43.3	58.3	66.6	76.2	12.6
Steering	82.2	90.3	84.6	88.5	76.4	78.7	70.1	73.2	50.0	51.7	72.7	76.5	5.0
SFWE (TL)	84.3	86.5	**88.3**	88.8	78.7	79.5	72.7	72.2	**51.7**	53.3	74.8	75.4	**0.8**
SFWE (SL)	82.6	90.2	85.4	88.5	81.1	80.3	71.7	75.1	50.0	61.7	74.2	77.9	4.7
SFWE (SL+TL)	**84.8**	87.1	86.2	88.5	**82.7**	85.0	**74.3**	74.8	48.3	48.3	**75.3**	76.8	2.0
Qwen3-4B													
Base	72.3	86.8	79.1	86.4	67.7	79.5	65.7	76.1	45.0	55.0	66.0	76.8	14.1
Steering	71.8	87.1	76.8	88.8	66.9	80.3	62.6	77.4	40.0	60.0	63.6	78.7	19.2
SFWE (TL)	83.8	87.7	**88.0**	88.5	**78.7**	78.7	74.3	77.7	**51.7**	46.7	73.5	75.4	2.5
SFWE (SL)	86.2	89.4	84.9	89.6	78.0	80.3	74.5	77.4	**51.7**	56.7	72.4	77.9	7.1
SFWE (SL+TL)	**86.4**	89.1	84.9	86.4	71.7	76.4	**76.1**	75.8	**51.7**	38.3	**74.1**	73.2	**1.2**

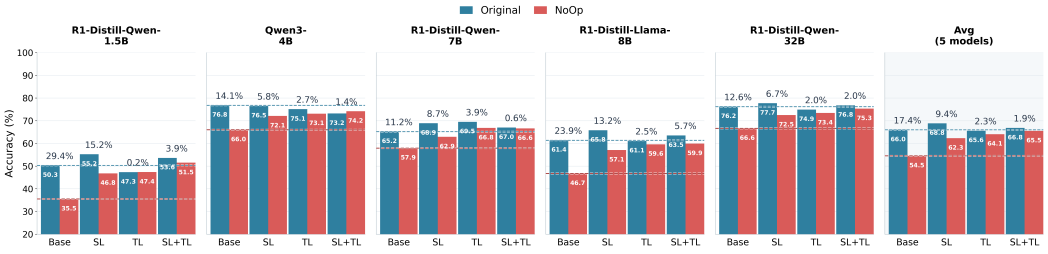

Figure 2: Original and perturbed (NoOp) accuracy averaged over five reasoning benchmarks. Solid blue and red bars denote Original and NoOp accuracy, respectively. Dashed blue and red lines mark the corresponding Base accuracies in each panel, providing references for comparing SL, TL, and SL+TL. The percentage above each bar pair is the relative gap, $|\text{Original}-\text{NoOp}|/\text{Original}\times 100\%$.

4.3 Robustness across Model Families and Scales

Figure 2 compares Orig and NoOp accuracy for all five models under the Base, TL, SL, and SL+TL settings, with results averaged over the five benchmarks. The trends from the main results persist across model families and scales. First, SL+TL achieves the largest average improvement on NoOp accuracy. Second, SL+TL produces the smallest relative Gap, bringing Orig and NoOp performance close to parity; TL ranks second, while SL leaves the largest residual gap of 9.4%, still reducing the Base gap from 17.4% by 46.0%. Third, none of the three editing variants decreases average Orig accuracy largely relative to Base, while SL and SL+TL improve it. Together, these results show that the main conclusions hold robustly across models ranging from 1.5B to 32B and across both Llama3.1-, Qwen2.5- and Qwen3-based model families.

5 Analysis

Vocabulary readout. We project each layer's direction through the unembedding and list the top-ranked tokens from its best-attested three-layer band (Table 2; Appendix C covers the other two models). The TL readout is dominated by additive and premise-introducing terms—*additionally*, *also*, *separately*, *given*, *another*, and their Chinese equivalents—as over-integration would predict. The SL readout is noisier but leans toward doubt and abandonment (*contradiction*, *somehow*, *nevertheless*, *abandon*), consistent with overthinking. Logit-lens readouts can be brittle (Belrose et al., 2023), especially for d_{SL}, which contrasts mean-pooled states rather than states at a shared next-token position, so we treat these lists as hypotheses and test them below.

Table 2: Top-ranked unembedding tokens of each direction for three of the five models (Table 6 gives the other two), from the three-layer band whose top tokens occur most often in the model's own traces. Bold marks tokens that fit the interpretation in the text; English glosses follow non-English tokens.

Direction	Model	Top-ranked tokens
TL	R1-Distill-Qwen-7B	**additionally**, **given**, **Also**, **additional**, **unrelated**
TL	Qwen3-4B	**additionally**, **there**, 另有 (**another**), **given**, **separately**
TL	R1-Distill-Qwen-32B	还有一个 (one more), 另有 (**another**), **there**, **also**, **separately**
SL	R1-Distill-Qwen-7B	矛盾 (**contradiction**), **somehow**, **unrelated**, forever, abandon
SL	Qwen3-4B	某种 (**some kind of**), Finally, 的重点 (key point), уж (indeed), ade
SL	R1-Distill-Qwen-32B	finally, 无论如何 (**nevertheless**), .LoggerFactory, ula, 마지 ("final" fragment)

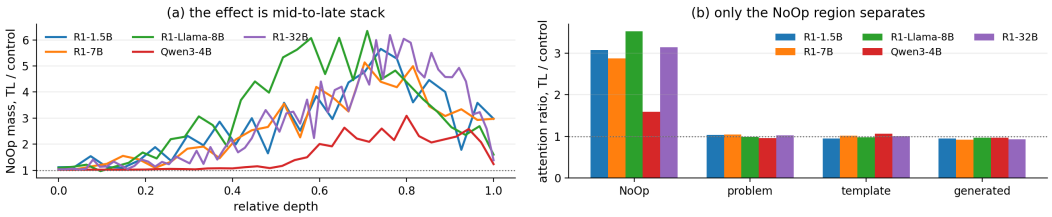

Figure 3: Attention from TL-active positions to prompt regions on NoOp Math500 ($n = 150$ traces per model; 152 for R1-Distill-Qwen-32B), relative to depth-matched control positions in the same trace. (a) Attention mass to the NoOp clause (standard errors over traces; labels give the ratio). (b) The NoOp-clause ratio by relative depth. (c) The ratio for each prompt region.

Token-level positions attend to the distractor. If the TL direction marks the uptake of an inserted premise, the positions where it is most active should attend to that premise. For each perturbed Math500 trace, we compare attention from the generated positions with the highest projection onto d_{TL} against control positions at matched depth in the same trace, since attention to the prompt decays with distance. On all five models, TL positions send 1.6–3.5$\times$ as much attention to the NoOp clause as controls (Figure 3a). The contrast is near parity in early layers and peaks at roughly 5–6$\times$ between 60% and 80% of depth on four models and about 3$\times$ on Qwen3-4B (Figure 3b). It is also specific to the clause: at the same positions, the ratios for the rest of the problem, the chat template and the generated prefix all lie between 0.92 and 1.06 (Figure 3c).

Editing reduces over-integration in the generated reasoning. To test whether the edit changes behavior, we give a judge (GPT-6-Luna) the unedited and SFWE traces for the same perturbed problem and ask which devotes less of its reasoning to the irrelevant condition (over-integration) and which spends less effort revisiting completed steps (overthinking). The judge sees both traces in full, is not told which is edited, and may call them comparable. Because verdicts depend strongly on presentation order, we judge every pair in both orders and count a verdict only when both pick the same trace. We sample 100 pairs per dataset, 20 per model, retaining 458 for over-integration and 466 for overthinking (Table 3).

For over-integration, the result is one-sided: the edited trace is rated less engaged with the distractor in 34.9% of pairs and more in 1.5%, winning 95.8% of separated pairs (95% Wilson CI $[91.6, 98.0]$). The rate is unchanged, 95.5% $[89.9, 98.1]$, among pairs with matching correctness, so the judge is not inferring behavior from the final answer, and *Less* exceeds *More* by 10 to 41$\times$ on every dataset. For overthinking, the evidence is weaker: the split is 42.3% against 27.7%, and the edited trace's win rate of 60.4% $[55.0, 65.6]$ falls to 54.7% $[48.0, 61.3]$ once correctness is held fixed, an interval that includes chance. We therefore take this as evidence that SFWE suppresses over-integration, but not clearly overthinking. This matches the edit's effect on trace statistics: repetition falls substantially on average, but mostly in a minority of runaway traces, while the median trace changes little.

Table 3: Pairwise judge comparison of unedited and SFWE-edited traces on the same perturbed problem (100 pairs sampled per dataset, 20 per model). *Pairs*: pairs judged for over-integration / overthinking. *Less*/*More*: both orders agree the behavior is less/more pronounced in the edited trace. *L/M*: their ratio. *Same*: rated comparable. *Ord.*: verdict flips when the traces are swapped. Other entries are percentages of judged pairs.

		Over-integration					Overthinking				
Dataset	Pairs	Less	More	L/M	Same	Ord.	Less	More	L/M	Same	Ord.
GSM-Symbolic	95 / 95	30.5	2.1	14.5	51.6	15.8	28.4	27.4	1.0	14.7	29.5
Math500	98 / 98	53.1	2.0	26.0	14.3	30.6	48.0	26.5	1.8	1.0	24.5
AMC	93 / 92	30.1	1.1	28.0	34.4	34.4	56.5	26.1	2.2	0.0	17.4
Gaokao-Math	93 / 95	44.1	1.1	41.0	29.0	25.8	40.0	30.5	1.3	2.1	27.4
AIME	79 / 86	12.7	1.3	10.0	35.4	50.6	38.4	27.9	1.4	0.0	33.7
All	458 / 466	34.9	1.5	22.9	32.8	30.8	42.3	27.7	1.5	3.6	26.4

6 Related Work

Robustness of LLM reasoning. Reasoning accuracy deteriorates under numerical substitutions (Mirzadeh et al., 2025), scenario reformulations (Huang et al., 2026) and distractor injection (Shi et al., 2023; Yang et al., 2025b), suggesting partial reliance on spurious features. Wnae adopt distractor injection, which Mirzadeh et al. (2025) show causes larger drops than changing names and numbers despite leaving the required solution unchanged.

Mitigating spurious correlations. Prior work mainly studies fixed-label NLU and short-output generation, using bias-aware training (Mahabadi et al., 2020), shortcut-token masking (Wang et al., 2022), data augmentation (Korakakis et al., 2025), causal reweighting (Zhou & Zhu, 2025), or Integrated-Gradients-based neuron pruning (Ali et al., 2025). These methods are not designed to capture spurious dependencies that emerge across chain-of-thought reasoning. For reasoning, LeaF uses confounding-token pruning and distillation (Guo et al., 2026), DeCoT selects reasoning paths through causal intervention (Wu et al., 2024), and Xie et al. (2025) apply correctness-perception distillation to chain-of-thought generation. Unlike these methods, which alter training or the generation process through distillation or causal intervention, SFWE addresses complex long-CoT reasoning with a training-free component-level edit that both suppresses spurious reliance and localizes interpretable spurious features to their associated model parameters.

Analyzing and modifying model behavior. Model behavior can be modified through activation steering (Turner et al., 2023; Rimsky et al., 2024) or parameter interventions for safety (Wei et al., 2024) and CoT-faithfulness analysis (Tutek et al., 2025). ThinkEdit (Sun et al., 2025) and Steer2Edit (Sun et al., 2026) derive behavioral directions from sequence-level activation contrasts and use them for component-level weight editing. SFWE follows a similar intervention pipeline, but targets on spurious-feature reliance in LLM reasoning through answer-preserving counterfactual interventions and augments sequence-level direction extraction with token-level contrasts to capture complementary behavioral granularities.

7 Conclusion

We present SFWE, a training-free weight-editing approach that combines mechanistic interpretability tools with answer-preserving counterfactual contrasts to identify and suppress spurious reliance in long-CoT reasoning. Across five open-weight models and five reasoning benchmarks, SFWE substantially improves robustness to irrelevant perturbations while largely preserving performance on original problems. Our analyses show that the extracted directions correspond to two recurring forms of spurious reasoning—over-integrating problem conditions and overthinking—and that suppressing these directions provides a mechanistic explanation for SFWE's robustness gains.

AI USE STATEMENT

We used generative AI tools to generate NoOp variants of the benchmark problems. We also used these tools to assist with paper searches, language polishing, code development and debugging, and manuscript review. The authors take full responsibility for the final manuscript, experimental results, and research artifacts, including all AI-assisted content.

REPRODUCIBILITY STATEMENT

Our experiments use publicly available benchmarks and open-weight models from the research community. The experimental settings, including NoOp data construction, inference configurations, and SFWE hyperparameters, are described in Section 4.1 to facilitate reproduction of our experiments.

REFERENCES

Ameen Ali, Shahar Katz, Lior Wolf, and Ivan Titov. Detecting and pruning prominent but detrimental neurons in large language models. *arXiv preprint arXiv:2507.09185*, 2025.

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. *Advances in Neural Information Processing Systems*, 37:136037–136083, 2024.

Nora Belrose, Igor Ostrovsky, Lev McKinney, Zach Furman, Logan Smith, Danny Halawi, Stella Biderman, and Jacob Steinhardt. Eliciting latent predictions from transformers with the tuned lens. *arXiv preprint arXiv:2303.08112*, 2023. doi: 10.48550/arXiv.2303.08112.

Qiguang Chen, Libo Qin, Jinhao Liu, Dengyun Peng, Jiannan Guan, Peng Wang, Mengkang Hu, Yuhang Zhou, Te Gao, and Wanxiang Che. Towards reasoning era: A survey of long chain-of-thought for reasoning large language models. *Science China Information Sciences*, 69(6):161101, 2026.

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

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, et al. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. URL `https://transformer-circuits.pub/2021/framework/index.html`.

Zhizhang Fu, Guangsheng Bao, Hongbo Zhang, Chenkai Hu, and Yue Zhang. Correlation or causation: Analyzing the causal structures of llm and lrm reasoning process. *IEEE Transactions on Audio, Speech and Language Processing*, 2026.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pp. 5484–5495, 2021.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Yiju Guo, Wenkai Yang, Zexu Sun, Ning Ding, Zhiyuan Liu, and Yankai Lin. Learning to focus: Causal attention distillation via gradient-guided token pruning. *Advances in Neural Information Processing Systems*, 38:24921–24948, 2026.

Shulin Huang, Linyi Yang, Yan Song, Shawn Chen, Leyang Cui, Ziyu Wan, Qingcheng Zeng, Ying Wen, Kun Shao, Weinan Zhang, et al. Thinkbench: Dynamic out-of-distribution evaluation for robust llm reasoning. *Advances in Neural Information Processing Systems*, 38, 2026.

Michalis Korakakis, Andreas Vlachos, and Adrian Weller. Mitigating shortcut learning with interpolated learning. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pp. 9191–9206. Association for Computational Linguistics, 2025. doi: 10.18653/V1/2025.ACL-LONG.450. URL https://doi.org/10.18653/v1/2025.acl-long.450.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, pp. 611–626, 2023.

Minpeng Liao, Chengxi Li, Wei Luo, Wu Jing, and Kai Fan. Mario: Math reasoning with code interpreter output-a reproducible pipeline. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 905–924, 2024.

Rabeeh Karimi Mahabadi, Yonatan Belinkov, and James Henderson. End-to-end bias mitigation by modelling biases in corpora. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pp. 8706–8716, 2020.

Mathematical Association of America. American invitational mathematics examination (AIME). https://maa.org/maa-invitational-competitions/, n.d.a. Accessed September 18, 2026.

Mathematical Association of America. American mathematics competitions (AMC). https://maa.org/student-programs/amc/, n.d.b. Accessed September 18, 2026.

Iman Mirzadeh, Keivan Alizadeh-Vahid, Hooman Shahrokhi, Oncel Tuzel, Samy Bengio, and Mehrdad Farajtabar. Gsm-symbolic: Understanding the limitations of mathematical reasoning in large language models. In *International Conference on Learning Representations*, volume 2025, pp. 94743–94765, 2025.

OpenAI. Introducing OpenAI o3 and o4-mini. https://openai.com/index/introducing-o3-and-o4-mini/, 2025.

Judea Pearl. *Causality*. Cambridge university press, 2009.

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, 2024.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H Chi, Nathanael Schärli, and Denny Zhou. Large language models can be easily distracted by irrelevant context. In *International conference on machine learning*, pp. 31210–31227. PMLR, 2023.

Chung-En Sun, Ge Yan, and Tsui-Wei Weng. Thinkedit: Interpretable weight editing to mitigate overly short thinking in reasoning models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 17012–17036, 2025.

Chung-En Sun, Ge Yan, Zimo Wang, and Tsui-Wei Weng. Steer2edit: From activation steering to component-level editing. *arXiv preprint arXiv:2602.09870*, 2026.

Kimi Team, Tongtong Bai, Yifan Bai, Yiping Bao, Jianfeng Cai, Xinyuan Cai, Peizhou Cao, Yuxuan Cao, Ziwei Chai, Y Charles, et al. Kimi k3: Open frontier intelligence. *arXiv preprint arXiv:2607.24653*, 2026.

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.

Martin Tutek, Fateme Hashemi Chaleshtori, Ana Marasović, and Yonatan Belinkov. Measuring chain of thought faithfulness by unlearning reasoning steps. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 9946–9971, 2025.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Tianlu Wang, Rohit Sridhar, Diyi Yang, and Xuezhi Wang. Identifying and mitigating spurious correlations for improving robustness in nlp models. In *Findings of the association for computational linguistics: NAACL 2022*, pp. 1719–1729, 2022.

Boyi Wei, Kaixuan Huang, Yangsibo Huang, Tinghao Xie, Xiangyu Qi, Mengzhou Xia, Prateek Mittal, Mengdi Wang, and Peter Henderson. Assessing the brittleness of safety alignment via pruning and low-rank modifications. *arXiv preprint arXiv:2402.05162*, 2024.

Junda Wu, Tong Yu, Xiang Chen, Haoliang Wang, Ryan Rossi, Sungchul Kim, Anup Rao, and Julian McAuley. Decot: Debiasing chain-of-thought for knowledge-intensive tasks in large language models via causal intervention. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14073–14087, 2024.

Hongyan Xie, Yitong Yao, Yikun Ban, Zixuan Huang, Deqing Wang, Zhenhe Wu, Haoxiang Su, Chao Wang, and Shuangyong Song. Mitigating spurious correlations between question and answer via chain-of-thought correctness perception distillation. *arXiv preprint arXiv:2509.05602*, 2025.

Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tieyan Liu. On layer normalization in the transformer architecture. In *International conference on machine learning*, pp. 10524–10533. PMLR, 2020.

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.

Minglai Yang, Ethan Huang, Liang Zhang, Mihai Surdeanu, William Yang Wang, and Liangming Pan. How is llm reasoning distracted by irrelevant context? an analysis using a controlled benchmark. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 13340–13358, 2025b.

Jason Zhang and Scott Viteri. Uncovering latent chain of thought vectors in language models. *arXiv preprint arXiv:2409.14026*, 2024.

Yuqing Zhou and Ziwei Zhu. Fighting spurious correlations in text classification via a causal learning perspective. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 4264–4274, 2025.

A Sparsity Calibration

Table 4 reports the mean regret $R_{g,m}(\tau)$ (Equation 13) over the full sweep on the first 100 Math500 problems and their NoOp variants. The selected values are $\tau^* = 25\%$ for TL–MLP, 5% for SL–MLP, 25% for TL–Attn and 100% for SL–Attn.

B Per-Unit-Type Breakdown of the Main Results

Table 5 expands the TL and SL rows of Table 1 into their three variants each.

C Vocabulary Readout for the Remaining Models

Table 6 completes Table 2 with the two models omitted there for space. The pattern is the one described in Section 5: the TL readout is a family of additive and premise-introducing terms, while the SL readout is harder to read.

Table 4: Mean regret $R_{g,m}(\tau)$ in accuracy points (Equation 13), averaged over the five evaluated models, on the 100 calibration problems.

	MLP neurons		Attention heads	
τ (%)	TL–MLP	SL–MLP	TL–Attn	SL–Attn
0.01	6.20	6.70	–	–
0.05	6.80	6.40	–	–
0.25	5.90	3.70	5.80	7.00
1	4.30	3.30	5.10	6.00
2	–	–	4.50	6.20
5	1.70	**2.00**	3.80	4.00
10	–	–	2.70	2.60
25	**0.80**	3.80	**0.80**	1.40
100	2.30	4.20	1.00	**0.70**

Table 5: Per-unit-type breakdown of the main results, expanding the TL and SL rows of Table 1. *-MLP* edits neurons only, *-Attn* heads only, *-Both* edits both unit types along that direction. Sparsities are calibrated on the first 100 questions of Math500: 25% for TL-MLP, 25% for TL-Attn, 5% for SL-MLP, 100% for SL-Attn. Bold marks the best NoOp result in each column.

Method	GSM		Math500		AMC		Gaokao-Math		AIME		Average	
	NoOp	Original	NoOp	Original	NoOp	Original	NoOp	Original	NoOp	Original	NoOp	Original
DeepSeek-R1-Distill-Qwen-32B												
Base	80.1	87.5	79.4	85.9	67.7	77.2	62.3	71.9	43.3	58.3	66.6	76.2
Steering	82.2	90.3	84.6	88.5	76.4	78.7	70.1	73.2	50.0	51.7	72.7	76.5
TL-MLP	83.1	85.8	86.7	88.8	78.7	77.2	71.7	71.2	46.7	51.7	73.4	74.9
TL-Attn	81.7	86.5	83.1	87.0	73.2	79.5	68.3	70.6	50.0	53.3	71.3	75.4
SL-MLP	81.5	88.4	85.2	88.5	79.5	80.3	70.4	75.1	45.0	51.7	72.3	76.8
SL-Attn	81.5	88.2	84.6	86.7	78.7	78.0	69.4	74.0	48.3	61.7	72.5	77.7
TL-Both	84.3	85.2	**88.3**	86.7	77.2	78.7	72.7	72.2	**51.7**	50.0	74.8	74.6
SL-Both	82.6	90.2	85.4	87.2	81.1	79.5	71.7	74.0	50.0	58.3	74.2	77.9
SL+TL	**84.8**	87.1	86.2	88.5	**82.7**	85.0	**74.3**	74.8	48.3	48.3	**75.3**	76.8
Qwen3-4B												
Base	72.3	86.8	79.1	86.4	67.7	79.5	65.7	76.1	45.0	55.0	66.0	76.8
Steering	71.8	87.1	76.8	88.8	66.9	80.3	62.6	77.4	40.0	60.0	63.6	78.7
TL-MLP	83.8	87.6	**88.0**	86.4	74.0	78.7	72.7	76.1	46.7	46.7	73.0	75.1
TL-Attn	79.5	87.7	84.3	88.5	**78.7**	76.4	73.5	77.7	**51.7**	46.7	73.5	75.4
SL-MLP	82.2	89.2	81.7	88.3	74.0	78.7	72.2	76.9	**51.7**	56.7	72.4	77.9
SL-Attn	86.2	89.3	84.9	89.6	78.0	79.5	74.5	77.4	36.7	46.7	72.0	76.5
TL-Both	82.0	86.7	86.4	85.6	78.0	76.4	74.3	74.8	40.0	45.0	72.1	73.7
SL-Both	83.8	89.4	84.1	88.0	70.9	80.3	74.3	76.9	38.3	50.0	70.3	76.9
SL+TL	**86.4**	89.1	84.9	86.4	71.7	76.4	**76.1**	75.8	**51.7**	38.3	**74.1**	73.2

D Data Efficiency Relative to RLVR

We compare SFWE with GRPO and JustRL runs initialized from DeepSeek-R1-Distill-Qwen-1.5B, reporting for each the checkpoint, up to step 200, with the highest NoOp average (Table 7). JustRL gains 3.4 points after 20,480 prompt presentations and 163,840 sampled responses, and GRPO 3.2 points after 15,360 prompt presentations and 122,880 responses. SFWE gains 16.0 points from 20 original–perturbed pairs and 100 calibration problems, without training.

Table 6: Top-ranked vocabulary of each direction for the two models not shown in Table 2. English glosses follow non-English tokens.

Direction	Model	Top-ranked tokens
TL	R1-Distill-Qwen-1.5B	**given**, **also**, **Additionally**, 还有一个 (one more), **mention**
TL	R1-Distill-Llama-8B	**given**, **Also**, **Additionally**, также (**also**), **separately**
SL	R1-Distill-Qwen-1.5B	Sure, tax, nothing, cross, latex
SL	R1-Distill-Llama-8B	yx, norm, allo, **maybe**, pending

Table 7: Comparison with RLVR on R1-Distill-Qwen-1.5B. NoOp accuracy (%) averaged over the five benchmarks (change from Base). RLVR rows use the best NoOp checkpoint up to step 200; prompts count cumulative presentations.

Method	Prompts	Responses	NoOp Avg
Base	0	0	35.5
JustRL	20,480	163,840	38.9 (+3.4)
GRPO	15,360	122,880	38.7 (+3.2)
SFWE	20 pairs	20 pairs	**51.5** (+16.0)