WATCH THE MODEL THINK: ON-POLICY EXTRACTION OF ACTIVATION STEERING VECTORS

Anonymous authors
Paper under double-blind review

ABSTRACT

When a model solves a problem on one attempt and fails it on the next, what separates the two is rarely the final answer token; it is the *trajectory* that reached it. Contrastive activation steering leaves that signal unused: CAA, SADI, RepE and ITI build their direction from *experimenter-supplied text*, recorded while the model *reads* rather than reasons. That choice also caps what the vector can express, since polarity must be written into the text, and a task judged only by *outcome* offers nothing to write it with. **ROAST** makes the trajectory itself the contrast: sample rollouts, let an outcome verifier split them into successes and failures, and contrast the reasoning that worked against the reasoning that did not. What changes is *which text the activations are read from*, and a *matched teacher-forced control*—rollouts, labels, answer text and pair counts held fixed, the trajectory alone stripped—points to the trajectory as what matters: on GSM8K at 0.6B the pairs alone buy $+0.12$ points and restoring the trajectories buys $+6.05$, the larger step on all six columns tested and the only one of the two that ever clears seed variance. On GSM8K, with token count and anchor held fixed, replacing the trajectory by an equal-length neutral prefix or by another question's reasoning falls *below* no intervention—worse than supplying no reasoning at all. The two corpora are also far apart geometrically: the two readings of one direction sit a median 74° and 72° apart at the two Qwen3 scales we probe, and correcting for a split-half null moves either by under 1°. Reading from rollouts calls for two corrections—keep the full difference vector, since Top-10% masking discards half its energy, and give each question one vote rather than one per pair—and only grouped aggregation still beats the unsteered baseline on GSM8K under 20% verifier noise. We rest the comparison on the benchmarks scored without an answer parser—GSM8K, MATH500 and IFEval—where ROAST is best in all six cells over two models, by up to $+9.7$ over no intervention, at $+6.4\%$ wall-clock and with no added context; on six parser-scored benchmarks it also has the best steering average on each of the three models compared. Across nine models (0.6B–122B, four families), ROAST improves on the unsteered model at every scale. `https://anonymous.4open.science/r/ORBIT-E014/`.

1 INTRODUCTION

Consider a model that solves a math problem on one attempt and fails it on the next. The two attempts may agree on the final answer token's *form* yet differ completely in substance: one carried the unit conversion through, the other dropped it three steps in. What separates success from failure lies not in the answer but in the *trajectory*, so a direction in activation space meant to push the model toward the successful mode should be read from there.

Contrastive steering methods read it elsewhere. CAA (Rimsky et al., 2024), SADI (Wang et al., 2025), RepE (Zou et al., 2025), and ITI (Li et al., 2024) all extract from supplied text and then deploy the resulting vector while the model writes its own continuation, so, as conventionally instantiated, the model's own reasoning never enters the vector.

A limit on expressiveness, not just a shift. Because the contrast is carried by *the experimenter's text*, polarity must be expressible *in* that text—a gold versus a wrong option for CAA, or, for RepE,

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.

Table 1: Scaling across capacity and architecture, including a 70B dense model and a 122B mixture-of-experts model. Accuracy (%) ± std; **bold** marks the steered result. The *sign* is consistent—all 16 cells improve—but not the *size*: these two tasks leave little headroom (six of eight unsteered IFEval scores already in 79.9–84.4) and demand the least reasoning of our nine benchmarks, the two conditions under which §5.3 predicts small gains; the large margins are on reasoning (Table 4). All cells use the standard configuration of §5.1 (grouped, L_2, N=1000; IFEval N=100 throughout).

Model	IFEval	TruthfulQA	Model	IFEval	TruthfulQA
Qwen3-0.6B	62.46 ±1.27	48.78 ±0.52	Qwen3-32B	80.31 ±0.14	85.39 ±0.21
+ ROAST	**63.27 ±0.28**	**49.91 ±0.23**	+ ROAST	**81.02 ±0.71**	**86.64 ±0.21**
Qwen3-8B	77.35 ±0.32	78.66 ±0.16	GLM4-32B	80.31 ±0.71	57.40 ±1.66
+ ROAST	**80.50 ±1.98**	**81.62 ±0.03**	+ ROAST	**81.11 ±0.71**	**69.51 ±3.83**
GLM4-9B	81.02 ±0.12	53.83 ±0.01	Llama-3-70B	81.02 ±0.44	85.20 ±0.61
+ ROAST	**81.82 ±0.42**	**60.10 ±0.17**	+ ROAST	**82.19 ±0.53**	**86.83 ±0.70**
Qwen3-14B	79.89 ±0.57	80.31 ±1.22	Qwen3.5-122B-A10B	84.31 ±0.29	90.40 ±0.22
+ ROAST	**81.30 ±0.28**	**82.32 ±0.44**	+ ROAST	**85.12 ±0.45**	**91.35 ±0.28**

an instruction, where the difficulty is plainest: no instruction reliably means "solve this correctly" rather than "solve this incorrectly." Correctness is a property of an outcome, not of a string one can supply in advance, so on GSM8K, MATH500, MMLU or IFEval these methods have no native pair construction and fall back to labels appended to questions (Table 2). Our method, **ROAST** (**R**eading **O**n-policy **A**ctivations from **S**elf-generated **T**rajectories), instead takes its two sides to be two of the model's own attempts at one question, one a verifier accepted and one it rejected. Nothing prevents handing a prior method a reasoning-bearing pair, and §5.3 does exactly that; what such an adaptation cannot supply is *content*, since these methods read a question plus a final answer, never the steps that produced it.

The distinction matters because steering is an unusually cheap control primitive: no gradients or weight updates, just a precomputed residual-stream direction (Turner et al., 2024; Rimsky et al., 2024; Zou et al., 2025; Stolfo et al., 2025). Write p_{tf} for the distribution over *texts a steering method reads* when the experimenter supplies them and p_{ar} for the one the model induces by decoding: existing methods estimate from a sample of p_{tf} and deploy on continuations from p_{ar}. This concerns *corpora*, not the forward pass—activations are fixed by the token sequence—so the two readings differ only in which text is read: a bare gold answer versus a full derivation. On GSM8K we measure median angles of 74° and 72° between them at 0.6B and 8B, a gap that survives correction against estimation noise (§3). Geometry ranks neither reading; §5.3 answers that behaviorally.

Why this matters most for reasoning. A teacher-forced pair encodes a question plus a *final answer* written by a human rather than derived by the model, so for a sentiment label little is lost; for a chain of arithmetic the loss is structural. The loss is in the activations, not only the text: the state preceding an answer the model has just reasoned its way to differs from the state after re-reading that answer bare. Prior steering work has accordingly been evaluated largely on classification and single attributes such as refusal or honesty (Rimsky et al., 2024; Li et al., 2024; Arditi et al., 2024); sampling rollouts removes this barrier, and reasoning is where our gains are largest (+9.7 on GSM8K at 0.6B). Swapping the corpus is not enough on its own: correlated Cartesian-product pairs overweight questions by difficulty and explode per-layer norms, so we add Continuous Soft Scaling (CSS) and Grouped Mean Normalization, one vote per question (§3).

Contributions. We argue that, for difference-of-means steering on outcome-verified tasks, the model's own verified rollouts are a better corpus to read from than supplied text, and supply the measurement, attribution and method to support this. **The claim (§2):** the trajectory is the signal prior contrastive methods forgo (Table 2). **The measurement (§3):** the two readings of one direction sit a median 72–74° apart, within 1° of that net of estimation noise, so the gap is a corpus effect. **The attribution (§5.3):** a matched teacher-forced control isolates the trajectory from the pairs, and on GSM8K three further controls separate reasoning from the context length it fills; the gain follows the trajectory on all six columns, and an equal-length prefix without that reasoning is worse than no trajectory at all. **The method (§4):** ROAST samples, verifies, contrasts and normalizes rollouts—no probes, masks or per-input machinery—and has the best steering average on each model where CAA, SADI and RepE all run, though not on every benchmark; grouped aggregation, adopted against the correlated pairs, is also what survives verifier noise.

Table 2: What carries the contrast, and what follows, *as each method is published*—not a claim that no adaptation exists (§5.3 tests one). *Native outcome pair* asks whether the method's own construction yields a pair, without an appended label, on a task judged only by outcome (GSM8K, MATH500, MMLU, IFEval). †RepE's construction applies syntactically—an instruction pair needs no gold answer—but no instruction expresses "solve this correctly" versus "solve this incorrectly," so the pair is not polarized by the outcome; we pair opposing instruction framings instead (App. B.7).

Method	Polarity source	Trajectory in pair	Native outcome pair
CAA	gold vs. wrong option	no (answer only)	no
RepE	honest vs. dishonest instr.	no (stimulus shared)	no†
ITI	curated true/false statements	no (statement only)	no
SADI	appended label	no (answer only)	no
ROAST	**outcome verifier**	**yes** (full rollout)	**yes**

2 Related Work

Contrastive direction methods. Activation steering intervenes on internal representations at inference time (Turner et al., 2024; Rimsky et al., 2024; Zou et al., 2025; Stolfo et al., 2025); many behavioral attributes are approximately linear in activation space, so simple geometric operations suffice (Park et al., 2024). A common recipe extracts a difference-of-means direction from paired executions: contrasting prompts (Turner et al., 2024; Liu et al., 2024; Subramani et al., 2022), appended answer options (CAA) (Rimsky et al., 2024), paired experimental and reference instructions over a shared stimulus read via the first principal component (RepE) (Zou et al., 2025), or instruction-present versus instruction-absent runs (Stolfo et al., 2025). ROAST belongs to this family and differs along one axis—*what carries the contrast* (Table 2): above, polarity is written into the supplied text as an option, a label, or an instruction, and a task judged only by its outcome offers no such text to write it into, so these methods must be adapted there (App. B.7). In ROAST the two sides are two of the model's own rollouts on the *same* prompt, split by an outcome verifier, so polarity comes from the outcome and the contrast is carried by the reasoning itself.

Sparsification and component selection. SADI (Wang et al., 2025) applies an input-conditional Top-k mask keeping the largest-magnitude dimensions; we measure that the top 10% of dimensions carry only about half the signal energy (§3.2), motivating CSS. Others intervene on localized components rather than the residual stream (Li et al., 2024; Chen et al., 2024; Doan et al., 2026), steer conditionally or per query (Weng et al., 2026), or edit weights (Fierro & Roger, 2026).

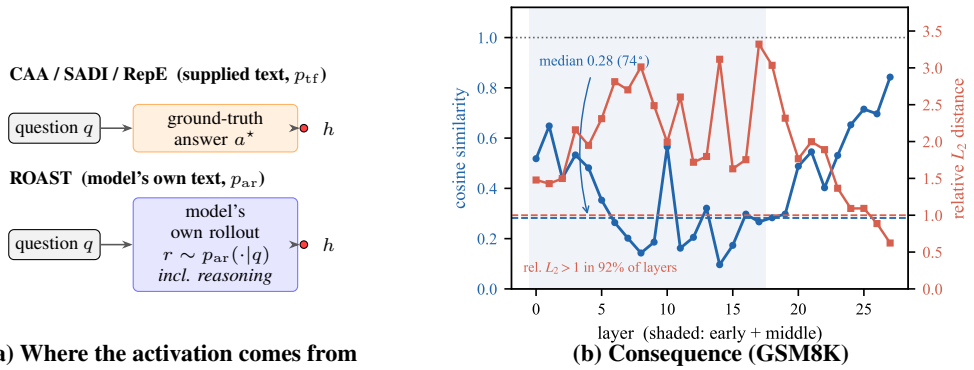
Scope of comparison, and its limits. Our baselines share ROAST's setting—training-free, residual-stream, direction estimated by contrasting activations—namely CAA, SADI, and RepE, all extracting on supplied text and so sitting on the axis we study; methods differing on further axes compose with ours rather than compete (App. B.7). Two comparisons we do *not* run bound the conclusion: training on the same verified rollouts (Zelikman et al., 2022; Yuan et al., 2023), whose data pipeline ROC shares, and a probe on the same activations (Alain & Bengio, 2017), the natural alternative estimator. We establish only that, among training-free contrastive directions with estimator and intervention held fixed, rollouts are a better corpus to read from than supplied text—not that a per-layer direction is the best use of a rollout pool. We also decline the *exposure bias* (Bengio et al., 2015; Ranzato et al., 2016) framing, which compounds through a decoding loop rather than one deterministic forward pass (App. E).

3 The Extraction–Deployment Mismatch

We motivate ROAST with three measurements, each targeting one default in the standard pipeline.

3.1 The extraction corpus is not the deployment corpus

What "on-policy" means here. Our claim concerns which corpus is sampled—p_{ar}, the texts the model itself produces, against p_{tf}, supplied ones—and nothing about decoding: activations are a deterministic function of the tokens run, so teacher-forcing a rollout the model generated returns exactly what generating it returned. Every quantity below is therefore attributable to the one axis that remains, a bare gold answer versus a derivation the model carried out (App. E).



(a) Where the activation comes from **(b) Consequence (GSM8K)**

Figure 1: The extraction–deployment mismatch. (a) What each method reads h from: prior methods, a curated ground-truth answer; ROAST, the model's own rollout, which contains the reasoning. The two rows differ in *text*, not in how the forward pass is run (§3.1). (b) Qwen3-0.6B/GSM8K, 64 questions, 8 rollouts each: $\cos(\mu_{\mathrm{tf}}, \mu_{\mathrm{ar}})$ has median 0.28 over early/middle layers and relative L_2 exceeds 1 in 92% of all layers; the same probe on Qwen3-8B gives a median angle of 72°. Panel (b) compares *two readings of one target direction*, not a good-vs-bad behavior gap, and it ranks neither.

Figure 1a makes the difference in what is read explicit; panel (b) quantifies it for Qwen3-0.6B on 64 GSM8K questions with 8 rollouts each. Over the first two thirds of layers $\cos(\mu_{\mathrm{tf}}, \mu_{\mathrm{ar}})$ has median 0.28 and falls as low as 0.10—74° at the median, up to 84°—while across all layers $\|\mu_{\mathrm{tf}} - \mu_{\mathrm{ar}}\|/\|\mu_{\mathrm{ar}}\|$ has median 1.92 and exceeds 1 in 92% of them, so the discrepancy is larger than either estimate's own norm. Qwen3-8B gives 72° with a larger relative L_2 median (2.69) in fewer layers (78%), and there the gap is concentrated in depth: the first seven layers stay at cosine 0.68–0.88 and it opens only from layer 8 on. Both probes use GSM8K, where trajectories are longest, and one model family, so we treat the angle as motivation for §5.3 rather than a general estimate; we expect it to shrink as trajectories shorten.

Is 74° large? A split-half null. The angle means nothing until one knows how far apart two estimates of the *same* direction fall, so we re-estimate each reading on disjoint halves of the 64 questions, over 200 splits. Two readings of one corpus agree to 5.3° (teacher-forced) and 13.6° (rollout) at 0.6B, 6.4° and 19.4° at 8B, against a 90° chance floor, so disattenuating the cross-corpus cosine moves it by 0.2° and 0.6° only—to 73.9° and 71.7°. The separation is a corpus effect, not an estimation artifact; the null does show the rollout reading to be the noisier of the two estimators, increasingly so with scale, by a margin far too small to account for the gap (App. B.5).

3.2 Discrete masking discards half the signal

Top-k masking—the sparsification at the core of SADI (Wang et al., 2025)—keeps the k largest-magnitude coordinates of the difference vector and zeroes the rest, presuming that the signal concentrates in a few dimensions. We measure the opposite: the top 10% of dimensions carry only 47–48% of the total L_2 energy, averaged over layers and consistent across both models and both tasks tested, so hard thresholding discards roughly half the direction (App. C.16), which motivates CSS (§4.2).

3.3 Naive aggregation weights questions by difficulty, not quality

Prior methods average pairwise differences directly, $v = \frac{1}{N}\sum_i (a_i^+ - a_i^-)$. Under rollout-based extraction this *naive aggregation* suffers from two biases. **Magnitude carries no quality signal:** a magnitude-weighted average is sensible only if larger contrastive vectors are better ones, and $\|\Delta h_q\|$ is essentially unrelated to *directional consistency*, the cosine with the consensus direction that governs steering quality under the linear representation hypothesis (Park et al., 2024) (App. Fig. 7). **Quantity imbalance:** because pairs form a Cartesian product $\mathcal{R}^+ \times \mathcal{R}^-$, a question with m correct and m incorrect rollouts contributes m^2 pairs from just $2m$ trajectories while a hard question may contribute one, so averaging over *pairs* weights questions by roughly the square of their rollout balance, a difficulty artifact growing with the rollout budget. The combined effect inflates the per-layer L_2 norm by one to two orders of magnitude with depth while grouping stays flat (App. Fig. 6); an estimator scaled by depth rather than signal is unusable, and §4.3 corrects it.

Algorithm 1 Rollout-based on-policy contrastive pair generation (ROC)

Require: Train set $\{(q_i, a_i^\star)\}_{i=1}^N$, rollout count n, temperature T
Ensure: Contrastive pairs $\mathcal{P}$
$\mathcal{P} \leftarrow \emptyset$
for each $(q_i, a_i^\star)$ **do**
  Sample $\{r_1, \ldots, r_n\} \sim p_{\mathrm{ar}}(\cdot \mid q_i)$ at temperature T
  $\mathcal{R}^+ \leftarrow \{r_j : \mathcal{V}(r_j, a_i^\star)\}, \quad \mathcal{R}^- \leftarrow \{r_j : \neg\mathcal{V}(r_j, a_i^\star)\}$
  if $\mathcal{R}^+ = \emptyset$ **or** $\mathcal{R}^- = \emptyset$ **then**
    continue {no pair exists (rates in App. C.1)}
  end if
  $\mathcal{P} \leftarrow \mathcal{P} \cup (\mathcal{R}^+ \times \mathcal{R}^-)$
end for
return $\mathcal{P}$

4 ROAST

ROAST has four steps, all offline except the last: (1) *sample* n rollouts per training question under p_{ar}, reading h at the final token of each; (2) *verify* and split into $\mathcal{R}^+/\mathcal{R}^-$, dropping questions with an empty side; (3) *aggregate*—difference of means within a question, normalize to a unit direction, average over questions, normalize again; (4) at inference, *add* αv per layer (App. Fig. 3). There are no trained probes, discrete masks or per-input adaptation, and one tuned hyperparameter, α.

4.1 On-policy contrastive pair generation (ROC)

For a training query q with reference answer $a^\star$, we sample n responses $\mathcal{R} \sim p_{\mathrm{ar}}(\cdot \mid q)$ at temperature T, split them by a verifier $\mathcal{V}(r, a^\star)$ into correct $\mathcal{R}^+$ and incorrect $\mathcal{R}^-$, and pair them as $\mathcal{P}_q = \mathcal{R}^+ \times \mathcal{R}^-$, skipping q if either side is empty (Algorithm 1).

The per-question contrast reads

$$\Delta h_q = \frac{1}{|\mathcal{R}^+|} \sum_{r^+ \in \mathcal{R}^+} h(q, r^+) - \frac{1}{|\mathcal{R}^-|} \sum_{r^- \in \mathcal{R}^-} h(q, r^-), \tag{1}$$

where $h(q, r)$ is the activation at the final token *of the rollout* r, so the whole trajectory sits in its context window. The verifier plays exactly the role ground-truth labels play for every contrastive baseline—partitioning positives from negatives, used *only offline*—so what differs is *what* gets labeled: curated text for prior methods, the model's own trajectories for ROAST, and many trajectories beat one, since averaging cancels the noise specific to any single path (§5.5).

4.2 Continuous Soft Scaling (CSS)

Rather than masking, we rescale the full vector: $v_{\mathrm{CSS}} = \mathrm{Norm}(\Delta h)$, using L_2 by default and comparing against max-norm (L_∞) in §5.3. Normalization limits activation outliers and puts all vectors on a common scale, making α comparable across layers and tasks (Huber & Ronchetti, 2009).

4.3 Grouped Mean Normalization

To remove both biases of §3.3 we adopt *one question, one vote*: normalize within each question, then average across questions, then normalize once more:

$$\bar{v}_q = \mathrm{Norm}\left(\frac{1}{|\mathcal{P}_q|} \sum_{(r^+, r^-) \in \mathcal{P}_q} \big(h(r^+) - h(r^-)\big)\right), \qquad v_{\mathrm{final}} = \mathrm{Norm}\left(\frac{1}{|Q|} \sum_{q \in Q} \bar{v}_q\right). \tag{2}$$

The target is thus the direction usable questions agree on, not the magnitude-weighted mean of Eq. (1), since $\|\Delta h_q\|$ carries no quality signal (§3.3); Norm returns zero below 10^{-8}, with no further weighting. Figure 2 gives the geometric view; empirically (App. Fig. 6) grouping keeps the pooled norm nearly constant across layers where naive pairwise averaging grows with depth, so α stays interpretable.

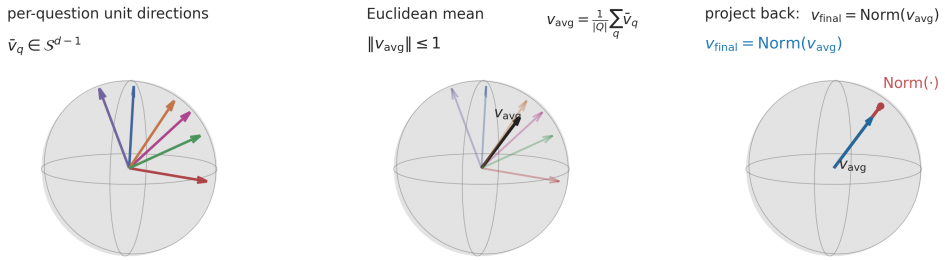


Figure 2: Geometric view of Grouped Mean Normalization. **Step 1:** per-question normalization collapses each question to a unit direction $\bar{v}_q$, discarding prompt-specific scale. **Step 2a:** their Euclidean mean v_{avg} lies inside the ball, its norm measuring directional agreement. **Step 2b:** the outer normalization projects it back to the sphere, so α alone controls strength.

Why group? Pairs from one question are *not* i.i.d.: formed as a Cartesian product over shared rollouts, they repeat the same semantic cues and failure modes. Two propositions (App. A.4) say what that costs the naive aggregator: it is a *weighted* average biased toward questions that happen to yield many pairs (Prop. 1), and under an explicit intra-question correlation assumption its variance does not vanish as N grows, whereas the grouped estimator's is $O(1/|Q|)$ (Prop. 2). What the propositions provide is a testable prediction rather than a guarantee—an estimator whose variance does not vanish should be fragile to label noise—which §5.4 tests.

4.4 Intervention

At inference we add the vector at layer l, component c (the MLP's post-activation output, before its down-projection), with strength α, at the last position of every forward pass—the final prompt token at prefill, then each generated token:

$$h'^{(l,c)} = h^{(l,c)} + \alpha \cdot v_{\text{final}}^{(l,c)}. \tag{3}$$

Vectors are layer- and task-specific (App. C); deployment is one elementwise addition per layer, with no extra forward pass and no added context, so inference cost matches CAA and SADI (+6.4% wall-clock, Table 7).

5 Experiments

5.1 Setup

Models. Qwen3-0.6B/8B/14B/32B (Yang et al., 2025), GLM4-9B/32B (GLM et al., 2024), Gemma3-1B (Team et al., 2025), plus Llama-3-70B and a Qwen3.5-122B-A10B mixture-of-experts model—nine models, four families.

Datasets. Nine benchmarks spanning truthfulness (TruthfulQA; Lin et al., 2022), instruction following (IFEval; Zhou et al., 2023), sentiment (SST2, SST5; Socher et al., 2013), reasoning (GSM8K, Cobbe et al., 2021; MATH500, Hendrycks et al., 2021b), and commonsense/NLI (MMLU, Hendrycks et al., 2021a; Winogrande, Sakaguchi et al., 2021; XNLI, Conneau et al., 2018). A steering set is drawn from training data and the remainder splits 20/80 into dev and test; all tasks are scored by one generate-then-extract harness, identical across methods, under the metrics of App. B–B.4.

Baselines. No intervention; few-shot ICL (Dong et al., 2024) with up to 100 exemplars (20 where context length binds); CAA (Rimsky et al., 2024); SADI (Wang et al., 2025); and RepE (Zou et al., 2025), whose reading operator we inject on *all* layers so that only the direction differs from ROAST, removing layer selection as a confound. CAA and SADI sweep $\alpha \in \{0.5, 1, 3, 5, 10, 50\}$ and SADI also top-$k \in \{5, 10, 50, 100\}$ per its paper, against ROAST's $\{0.5, 1, 3, 5\}$; App. B.7 documents negative-pair construction per baseline and task type.

ROAST configuration. Rollouts use n=64 (`max_tokens`= 256) for base-mode and n=8 (`max_tokens`= 1024) for chat-mode tasks, at T=0.8, top-p 0.9. Unless noted, we extract at

Table 3: Main results. Accuracy (%) ± std over three seeds. **Bold**/underline: best and second-best *steering* method per column within a model; few-shot ICL in *italics* is a reference only, consuming context at inference. All methods use one fixed configuration and N=1000 (ROAST: grouped, L_2); N=100 and nongrouped results are in Appendix C.7.

Model	Method	SST2	SST5	MMLU	TruthfulQA	Winogrande	XNLI	Avg
Qwen3-0.6B	No intervention	78.80 ±0.29	25.65 ±0.30	35.49 ±0.01	48.78 ±0.52	49.80 ±0.59	37.54 ±0.12	46.01
	Few-shot (100)	*84.03* ±0.38	*30.07* ±1.63	*40.20* ±0.05	*48.21* ±0.11	*49.49* ±0.08	*44.15* ±0.02	*49.36*
	CAA	78.55 ±0.25	**28.73 ±0.02**	35.81 ±0.01	35.19 ±0.00	49.80 ±0.00	37.13 ±0.01	44.20
	SADI	82.81 ±0.03	28.08 ±0.03	**38.16 ±0.04**	24.13 ±0.08	49.85 ±0.06	**43.92 ±0.02**	44.49
	RepE	82.37 ±0.55	28.51 ±0.61	37.60 ±0.28	47.05 ±0.40	50.44 ±0.39	43.16 ±0.55	48.19
	ROAST	**84.60 ±0.38**	27.91 ±0.14	38.15 ±0.01	**49.91 ±0.23**	**52.36 ±0.00**	41.72 ±0.01	**49.11**
Qwen3-8B	No intervention	86.68 ±0.21	38.12 ±0.58	66.99 ±0.01	78.66 ±0.16	64.29 ±0.03	69.37 ±0.01	67.35
	Few-shot (100)	*86.81* ±0.11	*50.42* ±0.31	*71.82* ±0.03	*72.66* ±0.18	*67.84* ±0.13	*70.84* ±0.03	*70.06*
	CAA	86.75 ±0.01	34.82 ±0.01	67.13 ±0.01	74.72 ±0.00	64.30 ±0.01	53.02 ±0.00	63.46
	SADI	87.25 ±0.01	39.39 ±0.01	66.93 ±0.02	72.28 ±0.15	61.34 ±0.01	69.56 ±0.01	66.12
	RepE	87.88 ±0.24	44.15 ±0.41	68.52 ±0.11	79.36 ±0.19	64.55 ±0.15	**69.58 ±0.07**	69.01
	ROAST	**88.47 ±0.01**	**48.17 ±0.15**	**69.11 ±0.02**	**81.62 ±0.03**	**64.64 ±0.01**	69.29 ±0.00	**70.22**
Gemma3-1B	No intervention	81.08 ±0.11	40.02 ±0.28	39.09 ±0.01	19.25 ±0.04	50.59 ±0.08	41.80 ±0.01	45.30
	Few-shot (100)	*84.05* ±0.00	*43.36* ±0.01	*37.58* ±0.01	*22.32* ±0.00	*47.71* ±0.00	*48.66* ±0.00	*47.28*
	CAA	82.69 ±0.10	40.15 ±2.26	39.51 ±1.34	21.33 ±0.45	50.27 ±1.34	**47.54 ±3.18**	46.91
	SADI	**85.10 ±1.86**	20.48 ±0.28	38.42 ±0.60	19.77 ±1.31	47.39 ±1.53	42.10 ±0.04	42.21
	RepE	83.36 ±0.98	41.20 ±0.63	39.71 ±0.51	21.02 ±0.74	51.29 ±0.88	45.88 ±1.12	47.08
	ROAST	84.31 ±0.58	**42.65 ±0.39**	**40.20 ±0.37**	**23.47 ±0.57**	**52.02 ±1.67**	44.29 ±0.11	**47.82**

the *last token of the sequence read*—for ROAST the rollout's last token, so prompt and full trajectory precede it; for teacher-forced baselines, the supplied answer's—apply the vector to MLP activations at every layer, and use L_2 CSS with grouped aggregation over N=1000 steering examples, except on IFEval, whose training split supports only N=100. The *only* tuned hyperparameter is α, selected on dev; every other choice is fixed across models and tasks, and the baselines share the same N and dev protocol. We report mean and standard deviation over three seeds $(33, 42, 52)$ under greedy decoding.

What is run where. We do not fill the 9×9 grid, since each cell would require re-sweeping SADI's 24-point grid; baselines run on three models $\times$ six benchmarks (Table 3) plus two $\times$ three reasoning benchmarks (Table 4), and at 70B and 122B we show only that ROAST helps. CAA and SADI have no defined pair construction on IFEval, whose target is constraint satisfaction rather than a gold string they could append, so there RepE is the only steering comparison (App. B.7).

5.2 Main results

Under a single fixed configuration (Table 3), ROAST attains the best average of any steering method on all three models, leading RepE—the strongest teacher-forced baseline—by $+0.92$, $+1.21$ and $+0.74$, and taking the most columns outright at every scale $(3/6, 5/6, 4/6)$. Against 100-shot ICL, a reference only since it consumes context at inference, ROAST is ahead on Qwen3-8B and Gemma3-1B and 0.25 behind on Qwen3-0.6B. RepE, unlike CAA and SADI, does not collapse on TruthfulQA or XNLI at 0.6B. Two caveats. No column here gives the baselines a native outcome pair, so the matched control of §5.3 does not cover them; the gap is evidence for the extraction axis, not for ROAST over these baselines' own configurations. And every benchmark here is scored through an answer parser, so these margins measure generate-and-parse accuracy, part of which may reflect answer extractability rather than selection (App. B.4); the trajectory evidence rests on the parser-free benchmarks of Table 4 in any case.

Are these margins significant? The comparison the design supports is per-model: ROAST beats RepE on $4/6$ columns at 0.6B and $5/6$ on each of Qwen3-8B and Gemma3-1B. A pooled Wilcoxon over the 18 (model, task) cells agrees ($14/18$ against RepE, $p = 0.009$) but overstates its confidence, since the cells are not exchangeable (App. C.4).

Where ROAST is weakest. On the two smallest models it loses individual columns to SADI and CAA, its Gemma3-1B margin over RepE is only 0.74, and few-shot edges out the 0.6B average. Every column it loses is base-mode, where rollouts are capped short, but base mode changes task type, rollout count and token cap together, so we do not read this pattern as causal (App. C.2).

Reasoning. Table 4 is where the extraction argument pays off most: $+9.66$ on GSM8K for Qwen3-0.6B, $+3.15$ on IFEval for Qwen3-8B, and the best steering result in every reasoning and instruction-following cell. It is also the cleanest controlled comparison available, since on GSM8K and MATH500

Table 4: Reasoning and instruction following. Accuracy (%) ± std over three seeds; **bold** is the best steering method, underline second. Few-shot (*italics*) is capped at 20 exemplars by context length and does not apply to IFEval, whose prompts are themselves the constraint.

	Qwen3-0.6B			Qwen3-8B		
Method	GSM8K	MATH500	IFEval	GSM8K	MATH500	IFEval
No intervention	48.86 ±0.19	33.59 ±0.47	62.46 ±1.27	92.61 ±0.66	56.09 ±0.78	77.35 ±0.32
Few-shot (20)	*46.21* ±0.17	*34.19* ±0.16	–	*92.46* ±0.75	*57.38* ±0.35	–
CAA	48.56 ±0.66	34.12 ±1.42	–	90.06 ±0.09	52.56 ±1.22	–
SADI	56.30 ±0.85	36.25 ±2.34	–	91.95 ±0.38	55.94 ±0.94	–
RepE	53.44 ±0.63	35.18 ±1.05	62.98 ±0.49	91.63 ±0.27	56.10 ±0.79	78.66 ±1.22
ROAST	**58.52 ±0.59**	**36.41 ±1.34**	**63.27 ±0.28**	**92.95 ±0.14**	**57.79 ±1.14**	**80.50 ±1.98**

Table 5: Component isolation with a matched teacher-forced control. Each row toggles one axis relative to the row above: extraction (TF → ROC), scaling (Top-k → full), aggregation (nongrouped → grouped, mean → max → L_2); rows 5–7 each replace row 4's nongrouped aggregation with one grouped reducer. Row 2 matches ROC on rollouts, labels, answer text, and pair count, stripping *only* the trajectory. Accuracy (%) ± std over the same three seeds $(33, 42, 52)$ as the main tables. All rows hold $N{=}100$ and come from one re-run of the ablation grid, so cells are comparable within this table but *not* with Table 3 or with the other single-seed $N{=}100$ tables (App. C.13); conclusions here are internal to $N{=}100$ (App. C.7).

		Qwen3-0.6B				Qwen3-8B			
#	**Variant**	SST2	MMLU	GSM8K	MATH500	SST2	MMLU	GSM8K	MATH500
1	Teacher-forced + Top-10%	79.57 ±0.17	37.78 ±0.19	48.74 ±0.10	34.51 ±0.28	86.24 ±0.20	67.38 ±0.05	89.40 ±0.65	53.34 ±0.26
2	+ matched ROC pairs (still TF)	–	37.82 ±0.17	48.86 ±0.54	34.81 ±0.32	–	67.51 ±0.17	89.94 ±0.19	53.64 ±0.24
3	ROC + Top-10%	81.95 ±0.15	39.41 ±0.32	54.91 ±0.46	36.02 ±0.36	88.18 ±0.48	68.22 ±0.17	91.80 ±0.21	56.17 ±0.05
4	ROC + full + nongrouped	82.29 ±0.37	39.57 ±0.19	55.39 ±0.85	36.13 ±0.12	87.75 ±0.26	68.68 ±0.11	91.67 ±0.23	57.41 ±0.15
5	+ grouped mean	83.08 ±0.12	**40.71 ±0.17**	57.17 ±0.51	36.30 ±0.21	88.16 ±0.20	68.92 ±0.16	92.42 ±0.50	57.23 ±0.54
6	+ grouped max-norm	83.70 ±0.10	40.15 ±0.29	57.05 ±0.22	36.32 ±0.36	88.46 ±0.16	69.03 ±0.09	92.30 ±0.29	57.33 ±0.16
7	+ grouped L_2 (ROAST)	**85.07 ±0.18**	40.58 ±0.19	**58.66 ±0.12**	**36.41 ±0.05**	**89.70 ±0.29**	**69.03 ±0.08**	**92.99 ±0.16**	**58.15 ±0.31**

the CAA/SADI negatives are *already* incorrect rollouts from ROAST's own pool, so negative difficulty is held constant and only the read context differs (App. B.7); App. C.18 shows a repaired mid-trajectory unit conversion, a fix an answer-only pair cannot encode.

5.3 Isolating the source of the gain: the trajectory, not the pairs

Does ROAST gain from *more* or *harder* pairs rather than from where the activations are read? Row 2 of Table 5 is the matched teacher-forced control: the *same* rollouts, labels, answer text and pair count per question as ROC, but each extraction reads the rollout's final answer *alone*, so rows 2 to 3 measure what the model's own thinking contributes on the six columns where a trajectory exists.

The split is lopsided: on GSM8K at 0.6B, adding ROC's pairs *without* their trajectories (row $1 \to 2$) buys $+0.12$ while restoring them (row $2 \to 3$) buys $+6.05$, and across all six columns the pair-only step stays within $[+0.04, +0.54]$ while the trajectory step spans $[+0.71, +6.05]$, larger on every one. Over three seeds every pair-only increment is *smaller* than the pooled standard deviation of the two rows it connects and every trajectory increment at least $2.5\times$ larger (App. C.3). One caveat remains: the size of the trajectory step does not track reasoning demand. We therefore claim only what the design tests—a pair-count account predicts steps $1 \to 2$ and $2 \to 3$ to be comparable, and on no column are they. Rows $3 \to 4$ isolate CSS and rows $4 \to 7$ aggregation: grouped L_2 beats nongrouped on all eight columns and is the best or tied-best reducer on seven, so the consistent factor is grouping rather than the reducer, *at this* N (App. C.7).

Is it the reasoning, or the room it takes up? Rows 2 and 3 differ in reasoning content, but also in context *length*. Three controls hold row 3's token count and final answer fixed and vary only how much of the model's own reasoning survives: a task-neutral *length-matched filler* (C1), a *mismatched trajectory* from another question (C2), and this rollout's own steps *shuffled* (C3). Every arm preserves the final answer string, so the anchor token is byte-identical, and positives and negatives are transformed alike, leaving pair counts, labels and answer text unchanged; all arms share one $\alpha{=}3$ (App. B.6).

The ordering is monotone in how much of this question's reasoning survives, over a 24.7-point span at fixed length, and *only* the intact trajectory beats no intervention. The decisive cells are C1 and C2:

Table 6: Extraction-context controls on GSM8K (Qwen3-0.6B, N=100, three seeds, grouped L_2, all arms at α=3). The five steered arms carry the *same* token count and a byte-identical anchor token, and differ only in how much of *this question's* reasoning the context retains. Configuration internal to this table (App. C.13).

	Extraction context	Accuracy	vs. no intervention
	No intervention	48.60	—
	Answer only (matched TF)	43.47	−5.13
C1	+ length-matched filler	**32.87**	**−15.73**
C2	+ mismatched trajectory	**36.67**	**−11.93**
C3	+ shuffled own trajectory	46.27	−2.33
	+ intact own trajectory (full ROC)	**57.60**	**+9.00**

Table 7: **Left: verifier noise.** Symmetric label flips in the $\mathcal{R}^+/\mathcal{R}^-$ partition (Qwen3-0.6B, N=100, single seed; grp = grouped, red = below the unsteered baseline (48.86 GSM8K, 48.78 TruthfulQA); App. C.5). **Right: runtime (min)**, Qwen3-8B/MMLU on one H20, at each method's own grid (4 points for ROAST, 6 CAA, 24 SADI; App. C.11).

	GSM8K		TruthfulQA	
Flips	grp	nongrp	grp	nongrp
0%	**58.52**	57.63	**49.22**	47.47
5%	58.01	55.90	48.74	45.31
10%	57.06	52.88	47.95	42.10
20%	54.83	46.71	46.02	37.88

	Stage			
Method	Bld	Swp	Test	**Tot**
None	–	–	12.5	12.5
CAA	0.32	19.8	13.2	33.3
SADI	0.39	60.6	13.3	74.3
ROAST	9.7	19.4	13.3	**42.4**

they are not merely weaker but actively harmful, landing 15.7 and 11.9 points *below* no intervention and 10.6 and 6.8 below the answer-only control that contains no reasoning at all. Supplying the same number of tokens without this question's reasoning is therefore worse than supplying no reasoning, which is not what a context-length account predicts. Under-steering cannot explain this, since a fixed α pulls every arm *towards* the unsteered baseline rather than 15.7 points below it.

5.4 Robustness to verifier noise

ROAST depends on a verifier, so we inject symmetric label noise into the $\mathcal{R}^+/\mathcal{R}^-$ partition before re-extraction, testing Proposition 2 (Table 7, left). Robustness tracks grouping: grouped stays above the unsteered baseline on GSM8K even at 20% flips, while nongrouped falls below it at 20% there and, on TruthfulQA, at every rate including 0%. Our verifiers err on well under 10% of rollouts, so these rates are pessimistic (App. C.5).

5.5 Design choices, scaling, and cost

Vectors are **task-specific**: each improves its own task while mismatched ones move accuracy by at most 0.3 points (App. C.6); every model in Table 1 improves, by up to +12.11.

Cost (Table 7, right). Building and tuning ROAST for Qwen3-8B/MMLU takes 42.4 min on one H20, below SADI's 74.3 and above CAA's 33.3: the larger one-time build buys the smallest dev sweep. Test-time overhead is at most +6.4%. Of the choices held fixed (App. C.8–C.12), two carry real margins: the MLP stream beats attention output by over 11 points on GSM8K, and dev-selected α sits next to a cliff—Qwen3-8B/MMLU falls from 69.25 to 44.67 between α=3 and 5—a fragility shared with CAA and SADI.

6 Conclusion

On reasoning tasks the trajectory separates success from failure, yet contrastive steering reads only experimenter-supplied text. Contrasting successful against failed rollouts moves extraction a median 72–74° from the teacher-forced reading, and a matched control attributes the gain to the trajectory context, not the pairs, on every column tested. ROAST is best suited to tasks where a verifier judges an outcome reached through visible steps. **Limitations.** It needs a verifier, assumes linearity (Park et al., 2024), and is untested against training on the same rollouts (App. D–E).

Reproducibility statement

Anonymized code is available at `https://anonymous.4open.science/r/ORBIT-E014/`. Appendix B specifies models, dataset splits and sizes, rollout and decoding hyperparameters, and the evaluation protocol, while Appendix B.7 documents baseline tuning grids and how contrastive negatives are constructed for each baseline and task type, including the RepE adaptation. All evaluation uses greedy decoding: main results and the component isolation of Table 5 are averaged over three seeds $(33, 42, 52)$, and the remaining ablations use a single seed (42), as noted in each caption. The geometric measurements of §3.1 share one extraction configuration (64 GSM8K questions, 8 rollouts each, seed 42), and the split-half null of Appendix B.5 averages 200 random half-splits of those same questions; angles from a separate rollout draw move by up to $1.7°$, which bounds their precision. Assumptions (A1)–(A4) and complete proofs of Propositions 1–2, together with a discussion of where the intra-question correlation assumption may fail, appear in Appendix A.4.

Ethics statement

ROAST steers model behavior at inference time without retraining, which can improve task accuracy and truthfulness but could in principle be redirected toward degrading behavior, since the method only requires a verifier defining the target outcome. The mechanism is the same as for existing activation steering methods and we introduce no new capability for producing harmful content; our experiments use public benchmarks only and involve no human subjects or private data. We note that steering vectors are task-specific and near no-ops off task (§5.5), which limits unintended spillover.

AI use statement

We used generative AI tools to assist with prose editing and LaTeX formatting of this manuscript. We did not use generative AI tools for research ideation, experimental design, or the derivation of the theoretical results. AI-assisted code was reviewed and tested by the authors, and all experimental numbers were produced by the authors' own runs. We have reviewed all AI-assisted work and take responsibility for the final content of this paper, including all text, claims, and artifacts.

References

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. In *International Conference on Learning Representations (ICLR) Workshop*, 2017.

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2015.

Zhongzhi Chen, Xingwu Sun, Xianfeng Jiao, Fengzong Lian, Zhanhui Kang, Di Wang, and Cheng-Zhong Xu. Truth forest: Toward multi-scale truthfulness in large language models through intervention without tuning, 2024. URL `https://arxiv.org/abs/2312.17484`.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021. URL `https://arxiv.org/abs/2110.14168`.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. XNLI: Evaluating cross-lingual sentence representations. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2475–2485, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1269. URL `https://aclanthology.org/D18-1269/`.

Toan Doan, Uyen Le, and Thin Nguyen. Causal activation steering via sparse mediation. In Vera Demberg, Kentaro Inui, and Lluís Marquez (eds.), *Findings of the Association for Computational Linguistics: EACL 2026*, pp. 1079–1097, Rabat, Morocco, March 2026. Association for Computational Linguistics. ISBN 979-8-89176-386-9. doi: 10.18653/v1/2026.findings-eacl.57. URL https://aclanthology.org/2026.findings-eacl.57/.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. A survey on in-context learning. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 1107–1128, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.64. URL https://aclanthology.org/2024.emnlp-main.64/.

Constanza Fierro and Fabien Roger. Steering language models with weight arithmetic, 2026. URL https://arxiv.org/abs/2511.05408.

Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Dan Zhang, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, Hao Yu, Hongning Wang, Jiadai Sun, Jiajie Zhang, Jiale Cheng, Jiayi Gui, Jie Tang, Jing Zhang, Jingyu Sun, Juanzi Li, Lei Zhao, Lindong Wu, Lucen Zhong, Mingdao Liu, Minlie Huang, Peng Zhang, Qinkai Zheng, Rui Lu, Shuaiqi Duan, Shudan Zhang, Shulin Cao, Shuxun Yang, Weng Lam Tam, Wenyi Zhao, Xiao Liu, Xiao Xia, Xiaohan Zhang, Xiaotao Gu, Xin Lv, Xinghan Liu, Xinyi Liu, Xinyue Yang, Xixuan Song, Xunkai Zhang, Yifan An, Yifan Xu, Yilin Niu, Yuantao Yang, Yueyan Li, Yushi Bai, Yuxiao Dong, Zehan Qi, Zhaoyu Wang, Zhen Yang, Zhengxiao Du, Zhenyu Hou, and Zihan Wang. Chatglm: A family of large language models from glm-130b to glm-4 all tools, 2024. URL https://arxiv.org/abs/2406.12793.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding, 2021a. URL https://arxiv.org/abs/2009.03300.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset, 2021b. URL https://arxiv.org/abs/2103.03874.

Peter J Huber and Elvezio M Ronchetti. *Robust Statistics*. Wiley, 2009.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model, 2024. URL https://arxiv.org/abs/2306.03341.

Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring how models mimic human falsehoods. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3214–3252, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.229. URL https://aclanthology.org/2022.acl-long.229/.

Sheng Liu, Haotian Ye, Lei Xing, and James Zou. In-context vectors: Making in context learning more effective and controllable through latent space steering, 2024. URL https://arxiv.org/abs/2311.06668.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 39643–39666. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/park24c.html.

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. Sequence level training with recurrent neural networks. In *International Conference on Learning Representations (ICLR)*, 2016.

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

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: an adversarial winograd schema challenge at scale. *Commun. ACM*, 64(9):99–106, August 2021. ISSN 0001-0782. doi: 10.1145/3474381. URL https://doi.org/10.1145/3474381.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In David Yarowsky, Timothy Baldwin, Anna Korhonen, Karen Livescu, and Steven Bethard (eds.), *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1631–1642, Seattle, Washington, USA, October 2013. Association for Computational Linguistics. URL https://aclanthology.org/D13-1170/.

Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul Christiano. Learning to summarize from human feedback. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Alessandro Stolfo, Vidhisha Balachandran, Safoora Yousefi, Eric Horvitz, and Besmira Nushi. Improving instruction-following in language models through activation steering, 2025. URL https://arxiv.org/abs/2410.12877.

Nishant Subramani, Nivedita Suresh, and Matthew Peters. Extracting latent steering vectors from pretrained language models. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 566–581, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-acl.48. URL https://aclanthology.org/2022.findings-acl.48/.

Mingjie Sun, Xinlei Chen, J. Zico Kolter, and Zhuang Liu. Massive activations in large language models, 2024. URL https://arxiv.org/abs/2402.17762.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, Gaël Liu, Francesco Visin, Kathleen Kenealy, Lucas Beyer, Xiaohai Zhai, Anton Tsitsulin, Robert Busa-Fekete, Alex Feng, Noveen Sachdeva, Benjamin Coleman, Yi Gao, Basil Mustafa, Iain Barr, Emilio Parisotto, David Tian, Matan Eyal, Colin Cherry, Jan-Thorsten Peter, Danila Sinopalnikov, Surya Bhupatiraju, Rishabh Agarwal, Mehran Kazemi, Dan Malkin, Ravin Kumar, David Vilar, Idan Brusilovsky, Jiaming Luo, Andreas Steiner, Abe Friesen, Abhanshu Sharma, Abheesht Sharma, Adi Mayrav Gilady, Adrian Goedeckemeyer, Alaa Saade, Alex Feng, Alexander Kolesnikov, Alexei Bendebury, Alvin Abdagic, Amit Vadi, András György, André Susano Pinto, Anil Das, Ankur Bapna, Antoine Miech, Antoine Yang, Antonia Paterson, Ashish Shenoy, Ayan Chakrabarti, Bilal Piot, Bo Wu, Bobak Shahriari, Bryce Petrini, Charlie Chen, Charline Le Lan, Christopher A. Choquette-Choo, CJ Carey, Cormac Brick, Daniel Deutsch, Danielle Eisenbud, Dee Cattle, Derek Cheng, Dimitris Paparas, Divyashree Shivakumar Sreepathihalli, Doug Reid, Dustin Tran, Dustin Zelle, Eric Noland, Erwin Huizenga, Eugene Kharitonov, Frederick Liu, Gagik Amirkhanyan, Glenn Cameron, Hadi Hashemi, Hanna Klimczak-Plucińska, Harman Singh, Harsh Mehta, Harshal Tushar Lehri, Hussein Hazimeh, Ian Ballantyne, Idan Szpektor, Ivan Nardini, Jean Pouget-Abadie, Jetha Chan, Joe Stanton, John Wieting, Jonathan Lai, Jordi Orbay, Joseph Fernandez, Josh Newlan, Ju yeong Ji, Jyotinder Singh, Kat Black, Kathy Yu, Kevin Hui, Kiran Vodrahalli, Klaus Greff, Linhai Qiu, Marcella Valentine, Marina Coelho, Marvin Ritter, Matt Hoffman, Matthew Watson, Mayank Chaturvedi, Michael Moynihan, Min Ma,

Nabila Babar, Natasha Noy, Nathan Byrd, Nick Roy, Nikola Momchev, Nilay Chauhan, Noveen Sachdeva, Oskar Bunyan, Pankil Botarda, Paul Caron, Paul Kishan Rubenstein, Phil Culliton, Philipp Schmid, Pier Giuseppe Sessa, Pingmei Xu, Piotr Stanczyk, Pouya Tafti, Rakesh Shivanna, Renjie Wu, Renke Pan, Reza Rokni, Rob Willoughby, Rohith Vallu, Ryan Mullins, Sammy Jerome, Sara Smoot, Sertan Girgin, Shariq Iqbal, Shashir Reddy, Shruti Sheth, Siim Põder, Sijal Bhatnagar, Sindhu Raghuram Panyam, Sivan Eiger, Susan Zhang, Tianqi Liu, Trevor Yacovone, Tyler Liechty, Uday Kalra, Utku Evci, Vedant Misra, Vincent Roseberry, Vlad Feinberg, Vlad Kolesnikov, Woohyun Han, Woosuk Kwon, Xi Chen, Yinlam Chow, Yuvein Zhu, Zichuan Wei, Zoltan Egyed, Victor Cotruta, Minh Giang, Phoebe Kirk, Anand Rao, Kat Black, Nabila Babar, Jessica Lo, Erica Moreira, Luiz Gustavo Martins, Omar Sanseviero, Lucas Gonzalez, Zach Gleicher, Tris Warkentin, Vahab Mirrokni, Evan Senter, Eli Collins, Joelle Barral, Zoubin Ghahramani, Raia Hadsell, Yossi Matias, D. Sculley, Slav Petrov, Noah Fiedel, Noam Shazeer, Oriol Vinyals, Jeff Dean, Demis Hassabis, Koray Kavukcuoglu, Clement Farabet, Elena Buchatskaya, Jean-Baptiste Alayrac, Rohan Anil, Dmitry, Lepikhin, Sebastian Borgeaud, Olivier Bachem, Armand Joulin, Alek Andreev, Cassidy Hardin, Robert Dadashi, and Léonard Hussenot. Gemma 3 technical report, 2025. URL https://arxiv.org/abs/2503.19786.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. Steering language models with activation engineering, 2024. URL https://arxiv.org/abs/2308.10248.

Weixuan Wang, Jingyuan Yang, and Wei Peng. Semantics-adaptive activation intervention for llms via dynamic steering vectors, 2025. URL https://arxiv.org/abs/2410.12299.

Zixuan Weng, Jinghuai Zhang, Kunlin Cai, Ying Li, Peiran Wang, and Yuan Tian. Finesteer: A unified framework for fine-grained inference-time steering in large language models, 2026. URL https://arxiv.org/abs/2604.15488.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Zheng Yuan, Hongyi Yuan, Chengpeng Li, Guanting Dong, Keming Lu, Chuanqi Tan, Chang Zhou, and Jingren Zhou. Scaling relationship on learning mathematical reasoning with large language models, 2023.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. STaR: Bootstrapping reasoning with reasoning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

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

Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, J. Zico Kolter, and Dan Hendrycks. Representation engineering: A top-down approach to ai transparency, 2025. URL https://arxiv.org/abs/2310.01405.

A METHOD DETAILS

A.1 THE PIPELINE END TO END

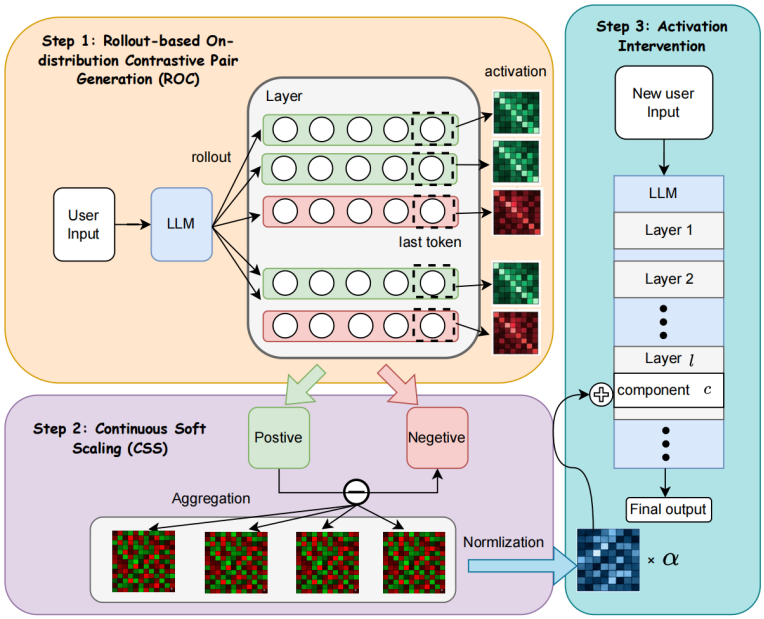


Figure 3: ROAST end to end. **Step 1 (ROC, §4.1):** the model produces n rollouts per question under p_{ar}; a verifier marks each correct (green) or incorrect (red), and the activation at the final token of each rollout is retained at every layer. **Step 2 (CSS + grouped aggregation, §4.2–4.3):** the difference of means within a question is normalized to a unit direction, those directions are averaged over questions, and the result is normalized once more. **Step 3 (intervention, §4.4):** at inference the vector, scaled by α, is added to component c of every layer l while a new input is decoded. Only Step 3 is online. (The figure labels Step 1 "on-distribution"; we call it *on-policy* in the text, following §E on the scope of that term.)

A.2 NORMALIZATION STRATEGIES FOR CSS

We consider two continuous rescalings of the difference vector.

- L_2 **normalization:** $\text{Norm}(\Delta h) = \Delta h / \|\Delta h\|_2$, projecting onto the unit hypersphere. This preserves direction exactly and normalizes total energy to unity.
- **Max normalization:** $\text{Norm}(\Delta h) = \Delta h / \|\Delta h\|_\infty$, scaling so the largest coordinate is 1. This prevents any single dimension from producing out-of-distribution activation magnitudes.

L_2 is our default; Table 5 compares both under grouped aggregation.

A.3 WHY NORMALIZE TWICE?

The two normalizations of Eq. 2 address distinct failure modes, which is why neither alone suffices. The *inner* one prevents loud prompts from dominating: activation magnitudes vary widely across prompts, so a global mean over all rollout pairs can be carried by a few high-amplitude questions, or by questions that happen to admit many pairs, yielding a vector that reflects amplitude outliers rather than the semantic consensus. The *outer* one decouples agreement from strength: after averaging unit directions $\|v_{\text{avg}}\| \leq 1$ by construction, approaching 1 when per-question directions align and 0 when they disagree, so using v_{avg} directly would tie the effective steering magnitude to incidental directional agreement and α would mean different things across datasets.

A.4 PROOFS

For a random vector $v \in \mathbb{R}^d$ write $\text{Var}(v) := \mathbb{E}\|v - \mathbb{E}v\|_2^2$, the trace of its covariance.

Setup. Let Q be the set of training questions. For $q \in Q$ the model yields n_q valid pairs with differences $\{v_{q,i}\}_{i=1}^{n_q}$, $v_{q,i} = h(r_{q,i}^+) - h(r_{q,i}^-)$. Let $N = \sum_q n_q$, $\mu_q = \frac{1}{n_q}\sum_i v_{q,i}$, and

$$v_{\text{naive}} = \frac{1}{N}\sum_q \sum_i v_{q,i}, \qquad \bar{v}_q = \text{Norm}(\mu_q), \qquad v_{\text{group}} = \text{Norm}\big(\tfrac{1}{|Q|}\sum_q \bar{v}_q\big).$$

Write $\mu_q^\star = \mathbb{E}[v_{q,i}]$ and $\mu^\star = \frac{1}{|Q|}\sum_q \mu_q^\star$.

Assumptions.

(A1) **Bounded differences:** $\|v_{q,i}\|_2 \le M$ for all q, i.

(A2) **Intra-question correlation:** samples within a question are semantically redundant, so $\text{Var}(\mu_q) = \Theta(M^2)$ rather than $\Theta(M^2/n_q)$. Rollouts conditioned on the same prompt tend to succeed or fail in stereotyped ways.

(A3) **Cross-question independence:** $\{\mu_q\}$, hence $\{\bar{v}_q\}$, are conditionally independent across q given the model.

(A4) **Non-degenerate consensus:** $\big\|\mathbb{E}[\frac{1}{|Q|}\sum_q \bar{v}_q]\big\|_2 \ge c > 0$.

Proposition 1 (sampling bias). *Conditional on $\{n_q\}$, the naive aggregator is a weighted average with weights $w_q = n_q/N$, so*

$$\mathbb{E}[v_{\text{naive}} \mid \{n_q\}] - \mu^\star = \sum_q \Big(\tfrac{n_q}{N} - \tfrac{1}{|Q|}\Big)\mu_q^\star,$$

which is non-zero whenever the $\{n_q\}$ are unequal and correlate with $\{\mu_q^\star\}$. Grouped aggregation enforces uniform weights $1/|Q|$ and removes this bias asymptotically.

Proof. We first rewrite the naive aggregator as a weighted average of per-question means. Conditioning on $\{n_q\}$ and grouping the inner sum by question,

$$v_{\text{naive}} = \frac{1}{N}\sum_q \sum_{i=1}^{n_q} v_{q,i} \qquad \text{(definition)} \tag{4}$$

$$= \frac{1}{N}\sum_q n_q\Big(\frac{1}{n_q}\sum_{i=1}^{n_q} v_{q,i}\Big) \qquad \text{(multiply and divide by } n_q\text{)} \tag{5}$$

$$= \sum_q \frac{n_q}{N}\,\mu_q \qquad \text{(definition of } \mu_q\text{)}. \tag{6}$$

So the weights are $w_q = n_q/N$, which sum to one but are non-uniform whenever the pair counts differ.

Next we take expectations. Since $\mathbb{E}[\mu_q] = \mu_q^\star$ and the weights are fixed once we condition on $\{n_q\}$,

$$\mathbb{E}[v_{\text{naive}} \mid \{n_q\}] = \sum_q \frac{n_q}{N}\,\mu_q^\star \qquad \text{(linearity of expectation)} \tag{7}$$

$$\mathbb{E}[v_{\text{naive}} \mid \{n_q\}] - \mu^\star = \sum_q \frac{n_q}{N}\,\mu_q^\star - \frac{1}{|Q|}\sum_q \mu_q^\star \qquad \text{(definition of } \mu^\star\text{)} \tag{8}$$

$$= \sum_q \Big(\frac{n_q}{N} - \frac{1}{|Q|}\Big)\mu_q^\star \qquad \text{(collect terms)}. \tag{9}$$

Each coefficient $n_q/N - 1/|Q|$ measures how far question q's weight departs from uniform. The sum vanishes only if the weights are uniform, or if the departures happen to cancel against the directions $\{\mu_q^\star\}$; in general it does not. Concretely, easier questions produce more correct *and* more incorrect rollouts, hence more pairs and larger n_q, so the residual systematically tilts toward easy-question directions.

Finally we show the grouped aggregator is free of this bias in the limit. As $n_q \to \infty$ we have $\mu_q \to \mu_q^\star$, and since $x \mapsto x/\|x\|_2$ is continuous away from the origin, the continuous mapping

theorem gives

$$\bar{v}_q \;=\; \mathrm{Norm}(\mu_q) \;\longrightarrow\; \frac{\mu_q^\star}{\|\mu_q^\star\|_2}. \tag{10}$$

Applying Slutsky's theorem to the average and then the outer normalization,

$$v_{\mathrm{group}} \;\longrightarrow\; \mathrm{Norm}\Big(\frac{1}{|Q|}\sum_q \frac{\mu_q^\star}{\|\mu_q^\star\|_2}\Big), \tag{11}$$

whose right-hand side depends only on $\{\mu_q^\star\}$ and not on $\{n_q\}$: the limit is invariant to how many pairs each question contributed. □

Proposition 2 (non-vanishing variance). *Under (A1)–(A4),*

$$\mathrm{Var}(v_{\mathrm{naive}}) = \Theta\Big(\tfrac{\sum_q n_q^2}{N^2} M^2\Big), \qquad \mathrm{Var}(v_{\mathrm{group}}) = O(1/|Q|).$$

The naive variance reduces to $\Theta(M^2)$ in the worst case $\max_q n_q \sim N$ and does not vanish with N, whereas the grouped variance contracts as $|Q|$ grows, regardless of $\{n_q\}$.

Proof. Part (i): the naive aggregator. Starting from the weighted-average form (6) and using cross-question independence (A3) to drop covariance terms,

$$\mathrm{Var}(v_{\mathrm{naive}}) = \mathrm{Var}\Big(\sum_q \frac{n_q}{N}\mu_q\Big) \qquad \text{(by (6))} \tag{12}$$

$$= \sum_q \frac{n_q^2}{N^2}\,\mathrm{Var}(\mu_q) \qquad \text{(independence, (A3))} \tag{13}$$

$$= \Theta\Big(\frac{M^2}{N^2}\sum_q n_q^2\Big) \qquad \text{(intra-question correlation, (A2)).} \tag{14}$$

The step to (14) is where (A2) enters: it asserts $\mathrm{Var}(\mu_q) = \Theta(M^2)$, i.e. averaging within a question does *not* reduce variance by a factor n_q, because the pairs are redundant.

It remains to bound $\sum_q n_q^2$. By Cauchy–Schwarz,

$$N^2 = \Big(\sum_q n_q\Big)^2 \le |Q|\sum_q n_q^2 \qquad\Longleftrightarrow\qquad \sum_q n_q^2 \ge \frac{N^2}{|Q|}, \tag{15}$$

with equality iff $n_q \equiv N/|Q|$; so even in the balanced case the variance is $\Theta(M^2/|Q|)$. In the imbalanced case, where one question dominates the pair budget,

$$\sum_q n_q^2 \;\le\; \Big(\max_q n_q\Big)\sum_q n_q \;=\; N\max_q n_q, \tag{16}$$

and this bound is tight when a single n_q carries most of the mass, giving

$$\mathrm{Var}(v_{\mathrm{naive}}) = \Theta\Big(M^2\,\frac{\max_q n_q}{N}\Big). \tag{17}$$

Setting $\max_q n_q \sim N$ yields $\Theta(M^2)$: the variance is bounded below by a constant independent of N, so growing the rollout budget does not make the naive estimator consistent.

Part (ii): the grouped aggregator. Write $S_Q = \frac{1}{|Q|}\sum_q \bar{v}_q$. Each summand lies on the unit sphere, so by the triangle inequality

$$\|S_Q\|_2 \;\le\; \frac{1}{|Q|}\sum_q \|\bar{v}_q\|_2 \;=\; 1, \tag{18}$$

and in particular each $\bar{v}_q$ has variance bounded by a constant. Using (A3) again,

$$\mathrm{Var}(S_Q) = \frac{1}{|Q|^2}\sum_q \mathrm{Var}(\bar{v}_q) \qquad \text{(independence, (A3))} \tag{19}$$

$$= O\Big(\frac{1}{|Q|}\Big) \qquad \text{(bounded summands, (18)).} \tag{20}$$

Note the contrast with (14): normalizing inside each question removes n_q from the expression entirely, so the rollout distribution can no longer inflate the variance.

One step remains, because the outer $\mathrm{Norm}(\cdot)$ is nonlinear and $\mathrm{Var}(v_{\mathrm{group}}) \neq \mathrm{Var}(S_Q)$ in general. Assumption (A4) places $\mathbb{E}[S_Q]$ at distance at least $c > 0$ from the origin, where $x \mapsto x/\|x\|_2$ is continuously differentiable with bounded Jacobian. The delta method therefore transfers the rate:

$$\mathrm{Var}(v_{\mathrm{group}}) = O\big(\mathrm{Var}(S_Q)\big) = O\left(\frac{1}{|Q|}\right), \tag{21}$$

which contracts to zero as the number of questions grows, for any $\{n_q\}$. □

How much these propositions actually claim. Four limits, stated because a reader who took them as proof that grouping is superior would be over-reading them.

(i) (A2) is load-bearing and close to the conclusion. The naive-side rate is not derived from the Cartesian-product structure; it follows from (A2), which *asserts* that within-question averaging does not contract. Were μ_q an average of n_q *independent* samples we would have $\mathrm{Var}(\mu_q) = M^2/n_q$ and $\mathrm{Var}(v_{\mathrm{naive}}) = M^2/N$, which *would* vanish, confining grouping's advantage to Proposition 1's bias. We argue (A2) is the right model here because pairs form a Cartesian product $\mathcal{R}^+ \times \mathcal{R}^-$ over a shared rollout pool, so a question with $|\mathcal{R}^+| = |\mathcal{R}^-| = m$ contributes m^2 pairs built from only $2m$ distinct trajectories, which repeat the same semantic cue and failure mode. The supporting evidence is indirect—the norm explosion of Figure 6 is consistent with non-vanishing variance and §5.4 tests the prediction—since we do not measure $\mathrm{Var}(\mu_q)$ against n_q directly, which would be the decisive test, and which the activations we already store would support at no additional sampling cost. Where rollouts are genuinely diverse, (A2) is conservative and grouping merely does no harm.

(ii) The two sides are not matched statements. The naive rate is a Θ, with a matching lower bound via Cauchy–Schwarz (15) and tightness of (16); the grouped rate is an O upper bound from boundedness of unit vectors (18) plus a delta-method transfer through the outer normalization. We establish neither that the grouped variance is $\Theta(1/|Q|)$ nor that its hidden constant is the smaller one, so the supported comparison is qualitative—"does not vanish in N" versus "provably contracts in $|Q|$."

(iii) The two estimators are not on a common scale. This is the sharpest form of (ii) and we state it separately because the proof makes it easy to miss. The bound (18) that delivers $O(1/|Q|)$ uses only that each $\bar{v}_q$ is a unit vector—a property grouping *imposes by construction* through the inner $\mathrm{Norm}(\cdot)$, before any question of aggregation arises. Meanwhile v_{naive} is unnormalized, with norm up to M, so a nontrivial part of the stated $\Theta(M^2)$-versus-$O(1/|Q|)$ gap is a difference in units rather than in how the pairs were pooled. The comparison that would isolate aggregation is $\mathrm{Norm}(v_{\mathrm{naive}})$ against v_{group}; both then live on the sphere, the trivial constant bound applies to each, and what remains is Proposition 1's bias plus whatever variance survives normalization. We do not carry out that analysis. Note that this cuts *against* the propositions and not against the method: the empirical comparison of §5.4 and Appendix C.7 is between two directions of equal norm, since ROAST normalizes at the end either way, and it is that comparison the paper relies on.

(iv) Variance is not accuracy. Neither proposition connects estimator variance to steering performance; that link is empirical, and the clean-data results in Appendix C.7—where two of six per-model averages favor nongrouped aggregation—show it is not tight. What the propositions predict is *where* grouping should matter most, namely when the partition is noisy, and §5.4 confirms it. That is the claim we defend.

B Implementation Details

B.1 Compute and inference stack

Unless noted, experiments run on two NVIDIA A100 (80 GB) GPUs, with rollout generation via vLLM (Kwon et al., 2023); the runtime table in Table 7 was measured on one H20 (141 GB). The 70B and 122B scaling runs use eight A100s with tensor parallelism. All nine models are instruction-tuned checkpoints; since the family names used in the tables (*e.g.* "Llama-3-70B") do not pin down a revision, exact checkpoint identifiers and prompt templates are listed in the released configuration files.

B.2 DATASETS AND SPLITS

For the main experiments we sample a **steering set** of $N = 1000$ examples from the training split, using the ground truth *only* to verify rollout correctness, never to build teacher-forced pairs. The remainder is split into a dev set (20%) for tuning α and a test set (80%) for reporting. Each seed induces a different assignment while preserving these sizes. The $N = 100$ rows in Appendix C.7 are ablations of steering-set size. IFEval is the one exception to $N{=}1000$: its training split supports only $N{=}100$, and every IFEval number in the paper—Tables 1 and 4 included—uses that setting for all methods.

Effective steering-set size. N counts questions *sampled*, not questions *used*. A question whose rollouts all land on one side of the verifier yields no pair and is discarded (Table 11), so the $|Q|$ that actually enters Eq. 2—and the $|Q|$ in Proposition 2's $O(1/|Q|)$—is smaller than N, by a factor that varies sharply across (model, task): on the diagnostic of Table 11 the usable fraction ranges from 14% to 63.5%. We do not report the realized $|Q|$ per cell of the main tables, which we now regard as an omission: it is the quantity the grouped estimator's variance depends on, and a reader cannot reconstruct it from N alone.

Table 8: Dataset splits. **Total eval** excludes the steering set. Steering-set size lists the $N{=}100$ ablation and the $N{=}1000$ main setting.

Dataset	Total eval	Steering set	Dev / Test	Source split
SST2	872	100/1000	174 / 698	dev
SST5	2,210	100/1000	442 / 1,768	test
MMLU	11,173	100/1000	2,234 / 8,939	test
TruthfulQA	717	100/1000	143 / 574	validation
Winogrande	1,267	100/1000	253 / 1,014	dev
XNLI	5,010	100/1000	1,002 / 4,008	test
GSM8K	1,319	100/1000	263 / 1,056	test
MATH500	400	100/1000	80 / 320	test
IFEval	441	100	88 / 353	—

B.3 METRICS

- **IFEval:** prompt-level *strict* accuracy—all verifiable constraints in a prompt must hold.
- **TruthfulQA:** MC1, single-answer multiple choice; a hit requires selecting the truthful option.
- **Reasoning:** exact match on the extracted final answer (GSM8K, MATH500).
- **Classification / NLI / commonsense:** accuracy after answer extraction (SST2, SST5, MMLU, Winogrande, XNLI).

All evaluations use greedy decoding for reproducibility, following the protocols of the original benchmark papers.

B.4 THE EVALUATION HARNESS, AND A CONFOUND IT CANNOT EXCLUDE

Every number in this paper is produced by one harness: the model generates under greedy decoding from a fixed per-task template, and a per-task parser extracts the predicted label, option, or final answer from that generation. Multiple-choice tasks (MMLU, TruthfulQA MC1, Winogrande) are scored the same way—by reading off the option the model *writes*—rather than by ranking option log-probabilities. The template, the parser, the decoding settings, and the test split are identical for every method in a table, which is what the comparisons require. Two consequences follow that a reader should weigh, and we state both because neither is visible from the tables.

Absolute numbers sit below published figures. Leaderboard results for these models typically use log-probability scoring, task-specific prompts, and few-shot contexts, all of which raise measured accuracy. Our unsteered Qwen3-8B reads 66.99 on MMLU and our unsteered Qwen3-0.6B reads 48.86 on GSM8K, both below the corresponding technical-report values. Cross-paper comparison of our absolute numbers is therefore not meaningful; within-table comparison is what the design supports.

A confound, and a preliminary diagnostic. Because scoring passes through a parser, a steering vector can raise accuracy in two distinct ways: by improving the selection, or by making the answer easier to extract—emitting a bare option letter instead of a hedged paragraph. We ran a first diagnostic separating them and report it here, with the caveat that it does not settle the question.

We rescore the three multiple-choice benchmarks by **log-probability ranking**: the option is chosen by argmax over the next-token log-probability of the option letters, so no generation and no parsing occur and an extractability improvement cannot register. Alongside it we report the **extraction-failure rate** of the generate-then-extract harness. Both are run unsteered and steered on the same test split and vectors.

Table 9: Preliminary log-probability rescoring (seed 42, α=3 fixed rather than dev-selected, N=1000 at 0.6B and N=300 at 8B). *gen* is the generate-then-extract protocol; *lp* is log-probability ranking, which has no parser; *fail* is the extraction-failure rate. This configuration is *not* that of Table 3—its *gen* margins do not reproduce that table and on 8B/MMLU differ in sign—so the table below should be read as a diagnostic of this run, not as a re-measurement of our reported margins.

Model	Task	gen	gen margin	lp	lp margin	fail (un. → ROAST)
Qwen3-0.6B	MMLU	35.10 → 42.30	+7.20	36.10 → 35.80	−0.30	2.6 → 0.7
	TruthfulQA	35.01 → 34.59	−0.42	34.73 → 34.73	0.00	0.0 → 0.0
	Winogrande	49.10 → 49.00	−0.10	48.90 → 48.70	−0.20	0.0 → 0.3
Qwen3-8B	MMLU	72.20 → 68.40	−3.80	74.90 → 75.00	+0.10	3.6 → 2.8
	TruthfulQA	70.29 → 70.29	0.00	66.25 → 66.11	−0.14	0.0 → 0.0
	Winogrande	65.20 → 51.30	−13.90	64.90 → 64.50	−0.40	0.5 → 23.8

What it does and does not establish. Under log-probability ranking every cell in this run lies within 0.4 points of zero. Two rows are informative about the mechanism. On Qwen3-0.6B/MMLU a +7.20 gen margin becomes −0.30 while the extraction-failure rate falls from 2.6% to 0.7%, which is what an extractability effect looks like. On Qwen3-8B/Winogrande the channel runs the other way: gen drops 13.9 points while lp moves −0.4 and the failure rate rises to 23.8%, so parser-mediated scores can move sharply in either direction without the underlying selection changing much. Absolute levels also differ between protocols—lp is more generous on MMLU and harsher on TruthfulQA—so only within-protocol margins are interpretable.

We stop short of reinterpreting Table 3 on this basis, for a reason internal to the run: it fixes α=3 instead of selecting it on dev, uses a single seed, and uses N=300 at 8B, and its *gen* margins consequently do not reproduce our reported ones—on 8B/MMLU it records −3.80 where Table 3 records +2.12. A diagnostic that does not reproduce the effect it is meant to explain cannot attribute that effect. What it does establish is that parser-mediated multiple-choice scores are sensitive to a channel unrelated to selection, which is enough to mark the mechanism behind our multiple-choice margins as open. The resolving experiment—rescoring at the main configuration, with dev-selected α and three seeds—we have not run, and until it is, the multiple-choice columns of Table 3 should be read as accuracy under this harness, with the reasoning-side interpretation unconfirmed.

Scope: the generative benchmarks are unaffected. This concern never applied to GSM8K, MATH500, or IFEval, whose metrics are exact match over the final answer and a programmatic constraint checker over the full generation, with no option parsing. Those are also where our largest margins lie (+9.66 GSM8K at 0.6B, +3.15 IFEval at 8B) and where the matched control of §5.3 and the context controls of App. B.6 are run. The trajectory claim rests on those benchmarks and is untouched by this section; what this section removes is the supporting role the multiple-choice columns were playing.

B.5 Split-half null for the extraction-corpus angle

§3.1 reports a median 74° between the teacher-forced and on-policy readings of one direction. That angle is uninterpretable on its own, since two estimates of the *same* direction from 64 questions also differ. We therefore measure the within-corpus reliability under an identical protocol: the 64 questions are split into disjoint halves of 32, each direction is re-estimated within each half, and the cosine between the two half-estimates is recorded per layer; we average over 200 random splits and report the median over the first two thirds of layers. The chance floor is the cosine between independent random directions in $\mathbb{R}^d$, distributed about 0 with scale $d^{-1/2}$. Only the retention of per-question

rows distinguishes this from the run behind Figure 1b, which it reproduces: cross cosine median 0.2738 (reported 0.28), minimum 0.100 at 84.3° (reported 0.10, 84°), and relative L_2 medians 1.911 and 2.653 (reported 1.92 and 2.69).

Table 10: Split-half null. GSM8K, 64 questions × 8 rollouts, median over the first two thirds of layers (IQR in brackets). *Within-TF* and *within-AR* re-estimate one corpus's direction on two disjoint halves of the questions; *cross* is the comparison of §3.1. The *disattenuated* row divides the cross cosine by $\sqrt{\rho_{\mathrm{tf}}\rho_{\mathrm{ar}}}$.

		Qwen3-0.6B		Qwen3-8B	
Comparison	**What it measures**	cos	angle	cos	angle
Random directions	chance floor ($d^{-1/2}$)	0.00±.031	90.0°	0.00±.016	90.0°
Within-TF (half A/B)	reliability of the TF reading	0.9957	5.3°	0.9938	6.4°
Within-AR (half A/B)	reliability of the AR reading	0.9720	13.6°	0.9434	19.4°
Cross TF vs. AR	the reported mismatch	0.2738 [.18, .47]	74.1°	0.3044 [.27, .67]	72.3°
disattenuated	mismatch net of noise	0.2775	**73.9°**	0.3145	**71.7°**

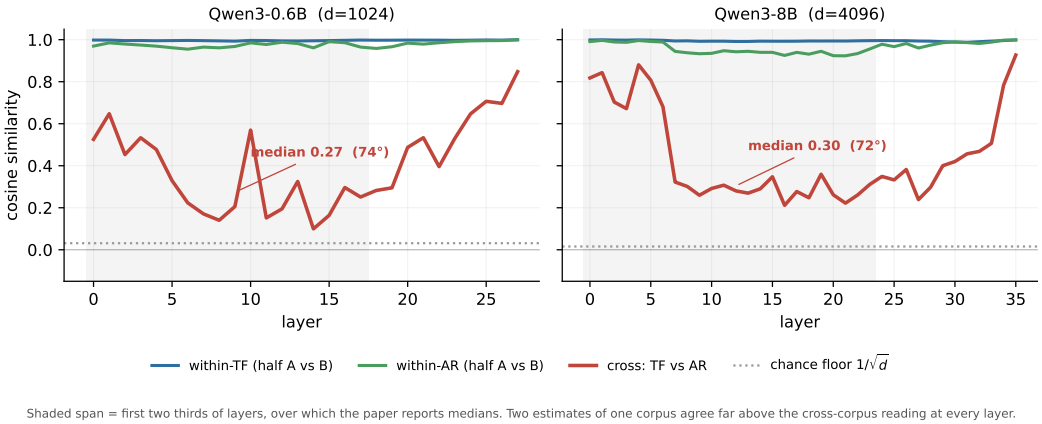


Figure 4: Per-layer view of Table 10. Two estimates of one corpus (blue, green) track each other near 1 at every layer, while the cross-corpus reading (red) runs far below them and far above the chance floor (dotted). Both readings converge again in the final layers, which is why medians are reported over the first two thirds throughout.

Three bands separate cleanly: the chance floor at 0.00, the cross-corpus cosine at 0.27–0.30, and the within-corpus reliabilities at 0.94–0.996. Correcting the cross cosine for attenuation therefore moves it by 0.2° and 0.6° only. Three qualifications. *(i)* The correction assumes the two readings' errors are uncorrelated; they are computed from the same questions, so positive correlation is plausible and would make this an over-correction, a bias that runs in favour of the reported gap. *(ii)* The corpus effect is layer-dependent where the noise floor is not: at 0.6B the cross cosine spans 0.100–0.648 over the reported layers while within-TF stays in 0.993–0.998; at 8B the first seven layers sit at 0.68–0.88 and the gap opens only from layer eight. *(iii)* As §3.1 states, none of this ranks the two readings. It establishes only that a real separation exists for §5.3 to be about.

B.6 Trajectory versus context length: construction of the three controls

Table 6 in the main paper reports these controls; this section gives their construction and their bounds. **C1, length-matched filler:** the reasoning span is replaced by a task-neutral span of equal token count. **C2, mismatched trajectory:** it is replaced by the model's own reasoning for a different, randomly drawn steering question, truncated or padded to matched length. **C3, shuffled trajectory:** this rollout's own steps are kept but permuted at the step boundary. Every control preserves the rollout's final answer string, so the anchor token is byte-identical across arms, and positives and negatives are transformed identically, so $|\mathcal{P}_q|$, the labels and the answer text never change.

Three bounds. *(i)* α is fixed at 3 rather than selected per arm. This is required for the comparison—per-arm tuning would confound the extraction context with how well each arm happened to be tuned—but it means no arm sits at its optimum. The direction of that concern runs against the controls rather than for them: under-steering pulls every arm *towards* the unsteered baseline, so it cannot

manufacture a 15.7-point drop below it. *(ii)* The decomposition is defined only where a trajectory step exists to decompose. On MMLU under this configuration there was none (all five contexts within 37.9–40.0, about one seed spread), consistent with base-mode rollouts carrying little trajectory (App. C.2); we therefore report the controls on GSM8K alone. *(iii)* One task and one model, at a configuration internal to that table.

B.7 Baseline configuration and negative construction

A fair comparison depends on how each baseline's negatives are built, which our earlier draft under-documented. We state it explicitly.

- **CAA / SADI, classification and multiple choice:** positives use the gold label appended to the question; negatives use a randomly chosen incorrect label. This is each method's standard construction.
- **CAA / SADI, reasoning:** negatives are incorrect model-generated rollouts drawn from *the same pool ROAST uses*, paired against a correct reference. Negative difficulty is therefore matched to ROAST on reasoning tasks, and the remaining difference is only that the baselines teacher-force the question plus final answer. This is why we regard the reasoning results as the cleanest controlled comparison.
- **RepE:** following the LAT procedure, the reading vector is the first principal component of paired stimulus differences, where each pair contrasts an experimental against a reference instruction over a shared stimulus. Since instruction polarity does not express "solve this correctly" versus "solve this incorrectly," outcome-verified tasks admit no native RepE pair; we therefore pair opposing instruction framings and inject the resulting direction via the linear-combination operator ($R' = R + \alpha v$) on *all* layers, sweeping $\alpha \in \{0.05, 0.1, 0.2, 0.25, 0.3, 0.5, 1.0, 1.5, 2.0, 3.0\}$. Injecting on all layers removes layer selection as a confound relative to ROAST, if anything favoring RepE.
- **IFEval:** CAA and SADI are not run, which is why their IFEval cells in Tables 1 and 4 are dashes; RepE is the only steering comparison on that column. The reason is the one §1 gives: both methods construct their pair by appending a gold or wrong answer string to the question, and IFEval's target is *constraint satisfaction*, for which no such string exists—a response either meets the constraints or does not, and there is no single correct output to append. Their construction is therefore undefined here, not merely inconvenient, which is the same expressiveness limit the paper argues for on outcome-verified tasks generally, in its sharpest form. RepE is unaffected because its pair is built from opposing *instructions* and needs no gold answer, which is why it alone extends to this column. We flag the consequence rather than leaving it to the dashes: IFEval carries one of ROAST's larger reported margins ($+3.15$ at 8B) against a single steering comparison, so it is the most thinly compared column in the paper.
- **Tuning grids:** CAA and SADI sweep $\alpha \in \{0.5, 1, 3, 5, 10, 50\}$; SADI additionally sweeps top-$k \in \{5, 10, 50, 100\}$ per its original paper. ROAST sweeps only α, over $\{0.5, 1, 3, 5\}$ (Appendix C.12). Every method reports its dev-set-best configuration under the identical protocol.

On classification the wrong-label negative is arguably easier than a model-generated one, which would favor the baselines' negatives being cleaner but less informative; the matched teacher-forced control in §5.3 isolates this by holding negatives fixed across the comparison.

Methods outside the comparison set. Several families of inference-time method differ from ROAST on axes other than the one we study, which is why they are not baselines: supervised probes over individual attention heads (Li et al., 2024; Chen et al., 2024) add training and change the intervention locus; refusal ablation (Arditi et al., 2024) targets a binary behavior rather than task accuracy; parameter-efficient tuning adapts weights at costs not directly comparable to a precomputed direction. All of them compose with ours—head selection, for instance, could perfectly well select over on-policy directions—so we regard them as complementary rather than competing.

Relation to exposure bias. Estimating on supplied text and deploying on generated text is superficially reminiscent of *exposure bias* (Bengio et al., 2015; Ranzato et al., 2016), and we state why we do not lean on the analogy. Exposure bias *compounds*: an error in a generated prefix conditions later steps, which is a property of the sequential decoding loop. A steering vector is read by a

single forward pass over an already-fixed token sequence, and that pass is deterministic in its input. What survives of the analogy is only the distribution-matching intuition—estimate from a corpus resembling the one you deploy on—and, unlike the classic remedies of scheduled sampling and sequence-level training, which modify *training*, a training-free estimator can act on it simply by changing which text it reads.

Methods we do not compare against. Two absences are worth naming because they bound what the main table can be taken to show. *(i) Training on the same rollouts.* Rejection-sampling fine-tuning and self-taught reasoning (Zelikman et al., 2022; Yuan et al., 2023) consume exactly the data ROC produces, by gradient update rather than by a residual-stream direction. We do not run them, so nothing here establishes that a per-layer direction is the best use of a verified rollout pool—only that, holding the estimator and the intervention fixed, rollouts beat supplied text as the corpus to read from. A reader deciding whether to *spend* the sampling budget, rather than how to use activations, should treat that comparison as open. *(ii) A probe on the same activations.* Fitting a linear classifier to the correct/incorrect rollout activations and steering along its normal (Alain & Bengio, 2017) varies the estimator while holding our corpus fixed—the complement of the experiment in §5.3, which holds the estimator fixed and varies the corpus. Running both would separate "the corpus is what matters" from "difference-of-means happens to suit this corpus." We ran only the latter.

C Additional Results

C.1 Discard-rate breakdown

Table 11: Fraction of steering questions with no valid contrastive pair (G=8 rollouts, N=200 prompts per cell, temperature 1.0, top-p 0.95). This is a standalone diagnostic run at a higher-entropy sampling setting than the main configuration (T=0.8, top-p 0.9; §5.1), which makes rollouts more likely to split, so the discard rates here are if anything optimistic relative to the main runs. It is a diagnostic of the regime, not a measurement of the realized $|Q|$ in any table.

Model	Task	All correct	All incorrect	Discarded	Usable
Qwen3-0.6B	SST2	33.5%	3.0%	36.5%	63.5%
Qwen3-0.6B	MATH500	14.0%	40.0%	54.0%	46.0%
Qwen3-8B	SST2	79.5%	6.5%	86.0%	14.0%
Qwen3-8B	MATH500	49.5%	26.0%	75.5%	24.5%

The two failure directions have opposite causes. A strong model on an easy task solves nearly every rollout, so pairs vanish for lack of negatives (Qwen3-8B/SST2). A weak model on a hard task fails nearly every rollout, so pairs vanish for lack of positives (Qwen3-0.6B/MATH500). ROAST operates in the middle band, where the model can produce the behavior but not reliably—which is also the band where steering is worth doing. Because a discarded question has an undefined difference-of-means, excluding it loses no signal; it only costs sample efficiency, and N and G can be raised cheaply offline.

C.2 Where ROAST loses, and why we do not read the pattern

§5.2 notes that every column ROAST loses is a base-mode column. The pattern is clean—base mode runs n=64 rollouts at `max_tokens`= 256, so rollouts are capped short and carry little trajectory, while every reasoning and instruction-following column, all of them chat-mode, is won outright—and it is exactly what the trajectory account predicts. We still decline to cite it as evidence, because the base-mode/chat-mode split moves task type, rollout count, and token cap together, so "the wins track trajectory length" and "the wins are on the reasoning tasks" are one statement in this design and cannot be separated. The isolating run that would separate them—one task evaluated at two token caps, everything else fixed—we did not do. The trajectory claim therefore rests on the matched control of §5.3, which varies only the extraction context, and not on this pattern.

C.3 Reading the matched control against seed spread

§5.3 states that the pair-only step (row 1 → 2 of Table 5) is inside seed noise on all six columns while the trajectory step (row 2 → 3) is outside it on all six. The comparison is each increment against

the pooled standard deviation $\sqrt{s_1^2 + s_2^2}$ of the two rows it connects, taken over the three seeds. For the pair-only step the ratios are 0.16 and 0.22 (MMLU, GSM8K at 0.6B), 0.71 (MATH500 at 0.6B), and 0.73, 0.80, 0.85 (8B), so no column reaches one pooled standard deviation. For the trajectory step they are 4.4, 8.5, 2.5 at 0.6B and 3.0, 6.6, 10.3 at 8B; the weakest, MATH500 at 0.6B, is $+1.21$ against a pooled ± 0.48. Two remarks on what this does and does not establish. Three seeds give a crude variance estimate, so these ratios are indicative rather than a test, and we quote no p-values from them; and the separation is a statement about the *gap between two extraction contexts*, not about the absolute accuracy of either, which depends on N and on the α selected on dev.

C.4 Pooled significance testing, and why we do not lead with it

§5.2 treats the per-model column win-rate as the primary evidence that ROAST's margins are consistent rather than driven by one cell. As a secondary summary we also pool Table 3, treating each (model, task) pair as one observation for 18 paired comparisons per baseline. ROAST wins $14/18$ against RepE (mean $+0.96$, one-sided Wilcoxon $p = 0.009$), $16/18$ against CAA ($+4.19$, $p = 0.0005$), and $13/18$ against SADI ($+4.77$, $p = 0.001$).

These p-values are optimistic and should not be read as the headline number. The test treats the 18 cells as exchangeable, whereas they are 3 models $\times$ 6 tasks: cells within a model share a checkpoint and a set of steering-set draws, and cells within a task share a benchmark and its idiosyncrasies, so the effective sample size is below 18 and the nominal level is anticonservative. A mixed-effects model with random intercepts for model and task would be the right instrument, but with three models it would itself rest on very few groups. We therefore report the pooled test for completeness and rest the consistency claim on the per-model win-rates ($4/6$, $5/6$, $5/6$), which make no exchangeability assumption.

C.5 Verifier-noise ablation: protocol

The table is in the main paper (Table 7); this section says what the test does and does not establish. At flip rate ϵ we corrupt the verifier's labels symmetrically (both directions at rate ϵ) before the $\mathcal{R}^+/\mathcal{R}^-$ partition is used for extraction, leaving everything downstream unchanged, and we re-extract both the grouped and the nongrouped estimator at each rate so the two arms differ only in aggregation. Flips are drawn *per rollout*, so a single flipped label follows that rollout into every pair it participates in; at a given ϵ this is a stronger perturbation than flipping pair labels independently would be. Because the flip is applied to the rollout labels before the partition, a question whose corrupted labels leave $\mathcal{R}^+$ or $\mathcal{R}^-$ empty is skipped by the same rule as in Algorithm 1, so noise shrinks the usable $|Q|$ as well as corrupting the surviving contrasts—a second channel, working against the grouped estimator, whose variance bound is $O(1/|Q|)$. Finally, α is held at the value selected on dev at $\epsilon{=}0$ rather than re-selected at each rate, so the degradation reported is that of a deployed vector whose verifier turns out to be noisier than assumed: the operationally relevant question, and the less favourable one, since re-tuning could only recover accuracy.

Two limits. Symmetric i.i.d. flips are the easiest noise model to specify but not the realistic one: a learned verifier errs in correlated, mode-specific ways, which we do not test (App. E). And the run is a single seed at $N{=}100$, so the evidence is the ordering of the two arms across rates, not any individual cell.

C.6 Cross-task interference and off-task retention

Table 12: Cross-task interference: apply task A's vector (rows) while evaluating task B (columns), as 0.6B / 8B accuracy (%). Each vector improves its own task (diagonal) while off-diagonal changes remain within the scale of the seed-to-seed variation reported in the main-paper tables. Single seed; the baseline row is repeated from the main-paper tables.

Steering vector	**SST2**	**GSM8K**	**MMLU**
None (unsteered)	78.80 / 86.68	48.86 / 92.61	35.49 / 66.99
SST2 vector	**85.24 / 89.94**	48.71 / 92.55	35.42 / 66.90
GSM8K vector	78.60 / 86.51	**58.52 / 92.95**	35.55 / 67.10
MMLU vector	78.74 / 86.60	48.60 / 92.48	**40.73 / 69.11**

Table 12 shows that each vector lifts its own task while leaving the other two close to unsteered performance at both scales; the behavioral test complements the activation-space diagnostic in Appendix C.17.

C.7 ROAST VARIANTS: GROUPING AND STEERING-SET SIZE

Table 13: Grouped versus nongrouped aggregation at two steering-set sizes. Each cell is grouped / nongrouped accuracy (%). The main paper reports the grouped, L_2, N=1000 configuration throughout; N=100 rows are included only to isolate steering-set size.

Model	SST2	MMLU	TruthfulQA	Grouped avg	Nongrouped avg
Qwen3-0.6B, N=100	**85.24** / 85.03	**40.73** / 37.32	**49.22** / 47.47	**50.03**	49.22
Qwen3-0.6B, N=1000	84.60 / **84.81**	**38.15** / 37.32	**49.91** / 49.39	**49.11**	48.52
Qwen3-8B, N=100	**89.94** / 89.11	67.71 / **69.06**	**81.18** / 80.14	68.79	**69.48**
Qwen3-8B, N=1000	88.47 / **89.11**	69.11 / **69.31**	**81.62** / 79.70	**70.22**	69.33
Gemma3-1B, N=100	**81.95** / 81.23	**39.87** / 39.68	19.43 / **19.86**	**47.26**	46.86
Gemma3-1B, N=1000	**84.31** / 83.10	**40.20** / 39.91	**23.47** / 22.77	47.82	**48.09**

Grouping helps in most settings but not uniformly: two of six per-model averages favor nongrouped aggregation, and individual cells go either way. We report this rather than selecting per cell. Two observations reconcile it with our claim. First, in the controlled sweep of Table 5, where everything except aggregation is held fixed, grouped L_2 beats nongrouped on all eight columns and the other two grouped reducers on seven of eight. Second, the clean-data comparison understates grouping's value: its decisive advantage appears under verifier noise (§5.4), where the nongrouped estimator falls below the unsteered baseline. Grouping buys robustness, and on clean data with a well-behaved rollout distribution there is simply less for it to fix.

C.8 ROLLOUT BUDGET AND INTERVENTION PLACEMENT

Table 14: Rollout budget and anchor position (Qwen3-0.6B), both at the ablation setting N=100 and so not comparable cell-for-cell with the main tables (App. C.13). **Left:** rollouts per question. **Right:** extraction anchor, token indices counted from the first response token; dashes were not run. Entries with ± are over three seeds, the rest single seed.

n	SST2	MATH500
2	80.72	36.25
8	84.66	36.41
64	85.24	36.31
128	**85.74**	**36.78**

Anchor	SST2	GSM8K	MATH500
Token 1	78.75 ±0.27	–	34.46 ±1.25
Token 32	83.14 ±2.81	–	34.11 ±0.18
Token 128	84.62 ±0.99	–	33.04 ±0.54
Sequence mean	83.41 ±0.13	**58.90 ±0.09**	**36.88 ±0.41**
Final token (default)	**85.24 ±0.29**	58.52±0.12	36.41 ±2.34

Table 14 sweeps the rollout budget and Table 15 the intervention site; Figure 5 shows the extracted direction converging as the budget grows, with per-layer norms stable throughout, which is why n=8 is our default despite n=128 scoring marginally higher.

Table 14 (left) reports the rollout-budget sweep. SST2 rises from 80.72 at n=2 to 84.66 at n=8 and only to 85.24 and 85.74 at n=64 and n=128: concave enough that n=8 captures most of the gain at an eighth of n=64's sampling cost, which is why we default to it despite n=128 scoring marginally higher. Only SST2 supports that reading. The MATH500 column spans 0.53 points end to end (36.25 → 36.78) and moves non-monotonically, which is inside the seed spread for that cell, so it neither confirms nor contradicts the concavity.

Table 15: Intervention placement (Qwen3-0.6B, single seed). **Left:** which component receives the vector. **Right:** which layers it is applied to. MLP at all layers is best.

Component	GSM8K	MATH500	IFEval	Layer scope	SST2	MATH500
MLP	**59.86**	36.41	**63.27**	All	**85.24**	**36.41**
Attention	48.34	34.69	57.93	First 5	81.38	32.34
MLP + Attn.	49.67	**36.88**	60.34	Last 5	81.02	33.13

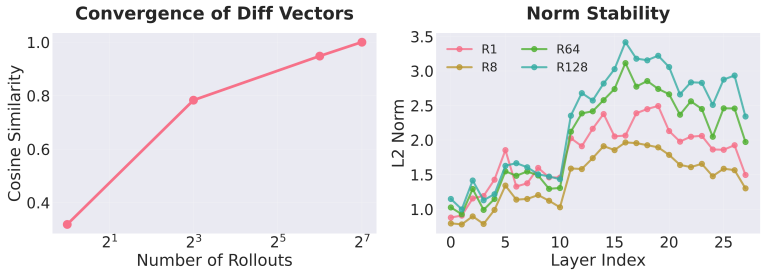


Figure 5: Rollout stability on Qwen3-0.6B. The extracted direction converges as the rollout budget grows, while per-layer L_2 norms remain stable across rollout budgets.

C.9 Anchor position

Table 14 (right) reports the anchor-position sweep, including the intermediate token positions (1, 32, 128) not discussed in the main text: accuracy rises monotonically with token position on SST2 but stays flat on MATH500, and none of the three matches the final-token default. A sequence-mean anchor is marginally better on reasoning (+0.38 GSM8K, +0.47 MATH500, both within seed variance) but clearly worse on classification (−1.83 SST2). The final token is therefore the best single default across all nine tasks, though a task-adaptive anchor is a reasonable refinement we did not pursue.

C.10 Steering strength

Table 16: Dev-set accuracy (%) across steering strengths α, single seed. Moderate values are best; too-large α can collapse behavior (Qwen3-8B MMLU at α=5).

Task	Unsteered	α=0.5	α=1.0	α=3.0	α=5.0
Qwen3-0.6B					
MMLU	36.12	37.74	38.32	39.12	**42.79**
SST2	77.59	82.18	85.06	**87.93**	86.78
SST5	24.21	29.86	**31.67**	31.00	28.73
TruthfulQA	50.35	52.45	52.45	52.45	**53.15**
Winogrande	46.64	53.36	53.75	53.36	**54.15**
XNLI	36.23	38.52	39.02	41.72	**50.60**
Gemma3-1B					
MMLU	40.06	42.39	42.84	**42.97**	42.26
SST2	87.36	87.93	87.93	**91.38**	89.08
SST5	42.53	43.44	43.67	45.25	**45.93**
TruthfulQA	13.99	22.38	23.08	**24.48**	**24.48**
Winogrande	52.96	53.36	53.36	**53.75**	53.36
XNLI	45.31	48.20	49.30	54.69	**58.28**
Qwen3-8B					
MMLU	67.46	67.64	67.99	**69.25**	44.67
SST2	89.08	90.23	91.38	**93.68**	87.36
SST5	37.56	39.59	38.91	40.50	**45.25**
TruthfulQA	79.72	**81.12**	**81.12**	80.42	79.72
Winogrande	65.22	66.40	**67.98**	66.80	66.01
XNLI	69.16	69.26	69.06	67.76	66.47

C.11 Runtime and cost accounting

The accounting is in the main paper (Table 7), measured end to end on one H20 for Qwen3-8B/MMLU. *Build* is direction construction: rollout generation plus extraction and aggregation for ROAST (9.7 min, of which sampling dominates), a single teacher-forced pass for CAA and SADI. *Sweep* is the dev grid, charged at each method's published grid—4 points for ROAST, 6 for CAA, 24 for SADI ($6\alpha \times 4k$)—and *Test* is one pass over the test split. Two consequences. All three steering methods add the same negligible test-time cost (13.2–13.3 against 12.5 unsteered, +6.4% at most), since deployment is one elementwise addition per layer with no extra forward pass and no added context. And the totals differ only offline, where the two costs have different characters: ROAST's build is

paid once per (model, task), whereas SADI's 24-point sweep is paid again on every re-tune, so the gap widens with the number of re-tunes rather than narrowing.

C.12 THE α SELECTION PROTOCOL, AND HOW FRAGILE IT IS

α is ROAST's only tuned hyperparameter, so the margins in Table 3 are only as trustworthy as its selection. We state the protocol and then the two ways it could be criticized, because Table 16 makes both visible.

Protocol. For each (model, task) we sweep $\alpha \in \{0.5, 1, 3, 5\}$ on the dev split (20% of the non-steering data; sizes in Table 8), select the argmax of dev accuracy, and apply that single value to the test split. Selection uses dev only, never test, and never the steering set. Table 16 tabulates the whole grid, and Table 7 charges the sweep at these four points against CAA's six and SADI's twenty-four. All three seeds share the protocol but select independently, so seed variance in the main tables includes variance in the selected α.

Concern 1: selection sits next to a cliff. Steering degrades non-monotonically in α, and the collapse can be abrupt: Qwen3-8B/MMLU reaches 69.25 at α=3 and falls to 44.67 at α=5—below unsteered (67.46) by more than 22 points. Dev selection picks α=3 here, i.e. the last value before the cliff. This is a genuine fragility of the method rather than of the protocol: a practitioner who tunes on a small dev set and guesses one step too far loses far more than steering ever gains. It also means the reported numbers depend on the grid containing a value in the safe region. We do not have a diagnostic that predicts the cliff location without evaluating past it, and we flag this as the practical risk of deploying ROAST (and, symmetrically, CAA and SADI, which share the α mechanism and a wider grid).

Concern 2: some optima land on the grid boundary. At 0.6B the dev optimum is α=5—the largest value swept—on MMLU (42.79), TruthfulQA (53.15), Winogrande (54.15), and XNLI (50.60); the same happens for Gemma3-1B on SST5 and XNLI, and for Qwen3-8B on SST5 (7 of the 18 tabulated cells in all). Two things follow, and they point in opposite directions. In ROAST's favor, the true optimum for those cells may lie outside the grid, so the corresponding numbers are if anything *under*-tuned. Against it, a boundary optimum means we cannot know how close those cells sit to the collapse documented above: at Qwen3-8B/MMLU the transition from best-on-grid to 22 points below unsteered happens in a single step from α=3 to α=5, so a cell selecting α=5 has not been shown to be on a plateau. We did not extend the grid, for two reasons: the sweep is the dominant cost in Table 7 and widening it asymmetrically would break the cost comparison; and extending only ROAST's grid while CAA and SADI keep theirs would favor us. The asymmetry that remains runs against ROAST: SADI sweeps 24 configurations to ROAST's 4.

Margins versus selection sensitivity. The honest reading is that ROAST's average margin over RepE (+0.74 to +1.21, Table 3) is smaller than the swing a one-step α error produces, so the margin should be read as "consistent across cells"—the 14/18 count and the per-model 4/6, 5/6, 5/6 column counts of §5.2—and not as a precise effect size. This is why we lead with the matched control of §5.3, where the compared variants share a selection protocol and the gap (+6.05 on GSM8K at 0.6B) is an order of magnitude above it.

C.13 CONFIGURATION DISCREPANCIES BETWEEN TABLES

Several ablations report a number for a cell that differs from the main tables, because they hold a different configuration fixed. We list every such case so none reads as an inconsistency.

- **GSM8K/Qwen3-0.6B: 59.86 (Table 15) versus 58.52 (Table 4).** The placement ablation sweeps the intervention component with α re-selected per component on dev, single seed; the main table fixes the default configuration and averages three seeds. The +1.34 difference is the value of per-component α re-selection at a single seed, not a better default.
- **Table 5 row 7 versus Table 3.** Row 7 is ROAST at N=100; the main tables use N=1000. Both average the same three seeds, so the residual differences (SST2/0.6B 85.07 versus 84.60,

GSM8K/0.6B 58.66 versus 58.52) are the effect of steering-set size alone, consistent with Appendix C.7.

- **Table 5 row 7 versus the single-seed $N{=}100$ tables.** Row 7 averages three seeds; Table 7 in the main paper and Tables 12, 13, and 15 here report seed 42 alone at the same $N{=}100$ setting, which is why the same configuration reads 85.07 ± 0.18 on SST2/0.6B in row 7 and 85.24 there. The single-seed values sit within about one standard deviation of row 7 on every shared cell; where a comparison depends on the spread we quote Table 5.
- **Configurations that beat our default.** Three exist in our own ablations, and we keep the default anyway: MLP+Attention on MATH500 (36.88 versus 36.41, Table 15); the sequence-mean anchor on GSM8K (58.90) and MATH500 (36.88, Table 14, right); and $n{=}128$ rollouts on both SST2 and MATH500 (Table 14). The first two are within the seed spread of the corresponding main-table cells (± 2.34 on MATH500) and reverse on other tasks—the sequence-mean anchor costs -1.83 on SST2—so tuning placement per task would be per-cell selection of the kind we avoid elsewhere; the third is a monotone compute-for-accuracy trade we decline in favor of the cheaper n (§5.5). A task-adaptive anchor or component is a real refinement we did not pursue.
- **Steering-set size.** Table 13 contains the only cells where a smaller steering set wins ($N{=}100$ beats $N{=}1000$ on SST2 and MMLU at 0.6B). We report $N{=}1000$ throughout regardless, since it is the better setting on average and selecting N per cell would inflate the results.
- **Cells that share a mean but not a spread.** ROAST on GSM8K/Qwen3-0.6B reads 58.52 ± 0.59 in Table 4 ($N{=}1000$) and 58.52 ± 0.12 in Table 14 (right, $N{=}100$); MATH500/0.6B likewise reads 36.41 with ± 1.34 in Table 4 and ± 2.34 in Table 14. These are separate three-seed runs at different steering-set sizes, not one run reported inconsistently; the means agreeing to two decimals across $N{=}100$ and $N{=}1000$ is a coincidence, and the spreads are what distinguish the runs. Where a comparison matters we quote the spread from the table the comparison is made in.
- **The $N{=}100$ default appears in two separate sweeps.** The default configuration is also the last row of the anchor sweep (Table 14, right) and row 7 of the component-isolation sweep (Table 5). The two sweeps were run separately over the same three seeds, so that cell appears twice with slightly different values (GSM8K/0.6B 58.52 ± 0.12 versus 58.66 ± 0.12; SST2/0.6B 85.24 ± 0.29 versus 85.07 ± 0.18; MATH500/0.6B 36.41 in both, at ± 2.34 versus ± 0.05). Every comparison we draw is within one sweep, where the compared variants share a run.
- **Which tables use which steering-set size.** The $N{=}1000$ main setting is reported in Tables 1, 3, and 4 only, with IFEval at $N{=}100$ throughout as noted above. Every ablation—Tables 5, 7, 14, and 15—holds $N{=}100$, so that varying the ablated axis is not confounded with steering-set size; cells are comparable within a table but not across this boundary. Table 16 is separate again: it reports *dev-set* accuracy, so its unsteered column (67.46 for Qwen3-8B/MMLU) differs from the test-set unsteered value (66.99, Table 3) for that reason and not because of N.
- **Table 1 uses the main configuration throughout.** An earlier version of that table reported 49.22 for ROAST on TruthfulQA/Qwen3-0.6B, which is the $N{=}100$ value of Table 13; the $N{=}1000$ value 49.91 ± 0.23 of Table 3 is the correct entry for that table's configuration and is what appears now. No other cell of Table 1 was affected.

C.14 GROUPED VERSUS UNGROUPED AGGREGATE NORMS

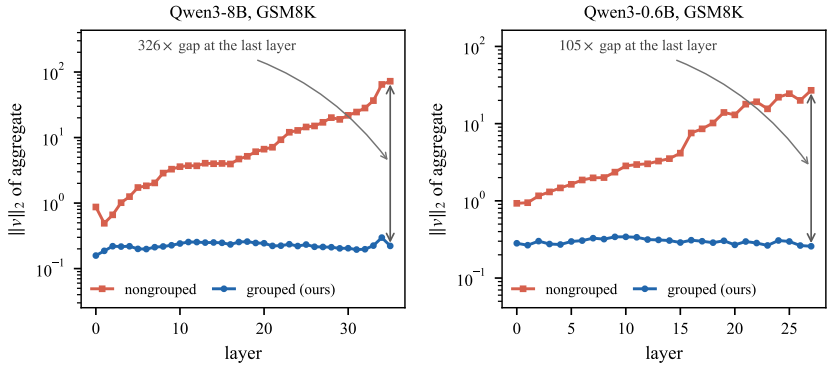


Figure 6: Per-layer L_2 norm of the aggregated steering vector (log scale). Ungrouped aggregation inflates with depth ($0.87 \rightarrow 72.4$ and $0.93 \rightarrow 27.0$) so magnitude is governed by depth, not signal; grouping stays flat, leaving α in control.

C.15 MAGNITUDE VERSUS DIRECTIONAL CONSISTENCY

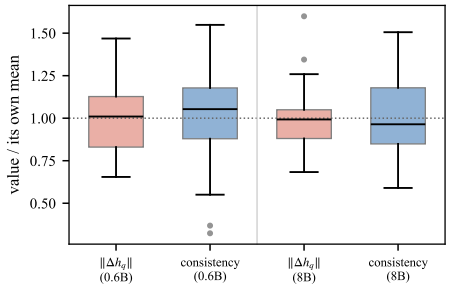


Figure 7: Per-question magnitude $\|\Delta h_q\|$ and directional consistency, each normalized by its mean (mid-network layer). Magnitude varies substantially across questions yet is only weakly and negatively correlated with consistency ($r = -0.10$ to -0.44), so a magnitude-weighted average adds variance without adding signal.

C.16 ENERGY RETAINED UNDER TOP-k MASKING

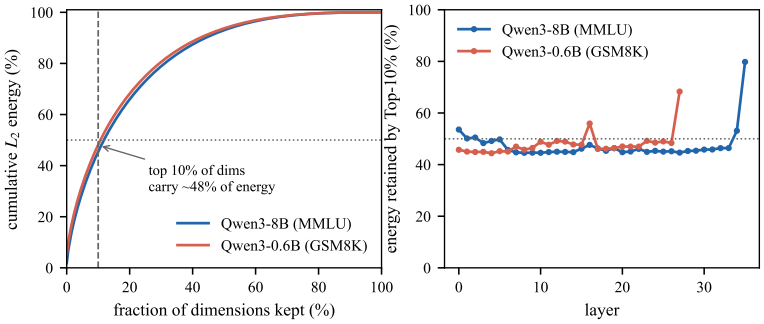


Figure 8: Discrete Top-k masking discards about half the signal. **Left:** cumulative L_2 energy of the contrastive difference vector as a function of the fraction of coordinates retained; keeping the top 10% retains only $\sim$48% of the energy. **Right:** the same quantity per layer, which stays near 50% throughout the network on both models.

The measurement is the reason ROAST rescales rather than masks. If the contrastive signal were concentrated in a few coordinates, zeroing the rest would be nearly lossless and would additionally suppress noise; the profiles above show it is not concentrated, so a Top-10% mask deletes about half the direction's energy and the deletion is spread over the whole network rather than confined to particular layers. That the retained fraction tracks the retained-coordinate fraction so closely also indicates the difference vector is not dominated by a handful of outlier dimensions, which is what magnitude-based selection implicitly assumes; the finding is consistent with the literature on "massive

activations" motivating magnitude-aware normalization over hard thresholding (Sun et al., 2024). Table 5, rows 3 $\rightarrow$ 4, gives the behavioral counterpart: replacing the mask with full-vector rescaling gains on six of eight columns, and the two exceptions (SST2 -0.43 and GSM8K -0.13, both at 8B) are inside one seed spread, so no column shows the mask to be preferable.

C.17 Layer, task, and component specificity

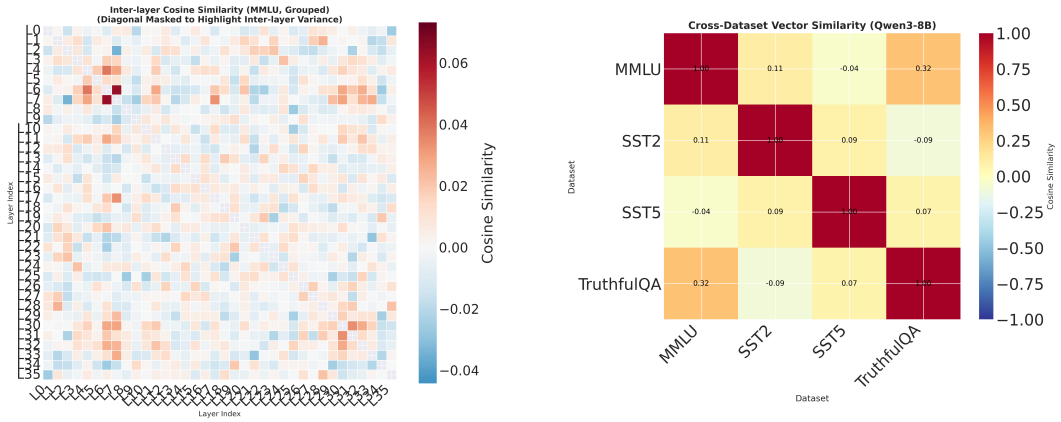


Figure 9: Task and layer specificity. **Left:** inter-layer cosine similarity of steering vectors (MMLU); distinct layers are nearly orthogonal (scale ± 0.06), so a single global direction would be inadequate and layer-wise intervention is necessary. **Right:** cross-dataset cosine similarity (Qwen3-8B); even closely related tasks such as SST2 and SST5 share little direction (< 0.1), indicating distribution-specific rather than category-level features.

The cross-task accuracy experiment above tests the same conclusion at the behavioral level, rather than only in activation space. Figure 10 adds the complementary MLP-versus-attention norm profiles.

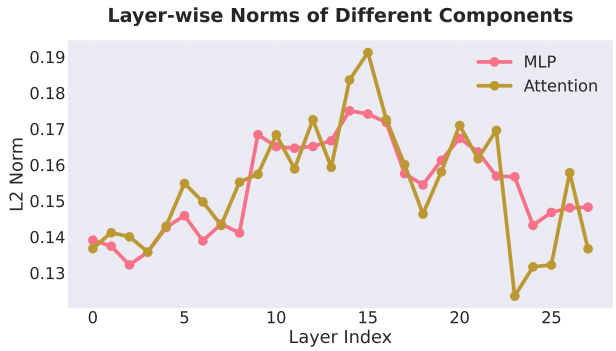


Figure 10: Per-layer norm profiles of steering vectors for MLP versus attention components, which have distinct magnitude structure across depth.

C.18 Qualitative examples

Chat mode: mathematical reasoning (GSM8K). *Question:* Thomas has \$1000 in \$20 bills. He loses 10 bills. He uses half of the remaining bills to pay for a bill. He then triples his money. He converts all his bills to \$5 bills. How many \$5 bills does he have?

Baseline response (incorrect):

1. Thomas starts with \$1000 in 20 dollar bills.
2. He loses 10 bills, so he has $\$1000 - 10 = 990$ dollars left. (Error: subtracts the number of bills from the total value.)
3. He uses half of the remaining bills to pay for a bill: $990/2 = 495$ dollars.
4. He triples the incorrectly computed amount: $495 \times 3 = 1485$.
5. He converts using the original bill denomination: $1485/20 \approx 74$.

Final Answer: 74

ROAST (ours) response (correct):

> To solve the problem, we need to track Thomas's money and bills step by step.
> 1. **Initial setup:** Thomas has \$1000 in 20-dollar bills. He loses 10 bills, so he has $\$1000 - 10 \times 20 = \$800$ remaining.
> 2. **After losing bills:** He uses half of the remaining money: $800/2 = 400$.
> 3. **Tripling his money:** $400 \times 3 = 1200$.
> 4. **Converting to 5-dollar bills:** $1200/5 = 240$.
> **Final Answer:** 240

Explanation. The baseline subtracts a bill count from a dollar amount, so all later arithmetic uses the wrong quantity. ROAST corrects this unit mismatch in the intermediate state before the subsequent arithmetic.

Generation mode: sentiment analysis (SST2). *Input:* `the piquant story needs more dramatic meat on its bones`

Baseline response (incorrect):

> positive
> The sentence expresses a positive sentiment, indicating that the story needs more substance, which implies a desire for improvement but frames it constructively.

ROAST (ours) response (correct):

> negative
> The sentence "the piquant story needs more dramatic meat on its bones" expresses a negative sentiment because it criticizes the story for lacking sufficient depth or substance ("dramatic meat").

Explanation. The baseline treats the request for improvement as positive feedback; ROAST reads "needs more ..." as criticism of insufficient depth.

D When to Use ROAST, and When Not To

The conclusion states the recommendation compactly; this section gives the decision rule our results support, including the cases where the answer is "do something else."

Use it when two conditions hold together. First, a verifier can label an outcome—exact match, a constraint checker, a multiple-choice key. Second, the model reaches that outcome through intermediate steps that a teacher-forced pair would not contain. Both hold on GSM8K, MATH500, and IFEval, and these are where the margins are largest ($+9.66$ GSM8K at 0.6B, $+3.15$ IFEval at 8B). The matched control of §5.3 says why the conjunction matters: the gain tracks the trajectory term on every column tested. We do not claim that term is ordered by task difficulty—at 0.6B it is marginally smaller on MATH500 than on MMLU—only that it is present wherever a trajectory exists and largest where trajectories are longest.

Prefer prompting when the trajectory is near-empty and the difficulty is output format rather than derivation. On SST2/SST5 a teacher-forced pair already carries nearly all of what the on-policy pair does, the ROAST-over-baseline margins shrink, and 100-shot ICL beats every steering method on the 0.6B average (Table 3). An exemplar demonstrates a format directly; one direction per layer cannot. If context budget is the binding constraint the ranking reverses, since ROAST consumes no context—but that is a deployment constraint, not an accuracy argument.

Do not use it when rollouts do not split—the two saturation regimes and their discard rates (up to 86%) are in App. C.1. Raising N and n recovers sample efficiency but cannot manufacture a pair where the model is deterministic in outcome, and this bound applies to any extract-from-behavior method.

Budget the risk of α. A one-step error in the strength grid can cost far more than steering gains—over 22 points at Qwen3-8B/MMLU (Appendix C.12)—so a dev split large enough to select α reliably is a precondition, not a detail. This risk is shared with CAA and SADI.

E EXTENDED LIMITATIONS

Linearity. We assume task-relevant concepts are approximately linearly separable in activation space (Park et al., 2024). This may fail for behaviors that are compositional or context-dependent, in which case a single direction per layer is the wrong functional form.

Rollout cost. ROC requires $O(nNL)$ decoding steps—about $140\,\mathrm{s}$ for a 32B model with $N{=}64$, $n{=}100$ on MMLU under vLLM. This is a one-time offline cost that leaves the per-token inference cost unchanged relative to other steering methods (Table 7), and ROAST remains effective at $n{=}8$ (§5.5), but CAA does not pay it.

Verifier dependence. ROC needs a verifier to partition rollouts. Benchmark metrics serve this role here; open-ended generation would require a learned reward model (Stiennon et al., 2020), whose own errors would enter the pipeline as label noise. §5.4 characterizes tolerance to such noise, and grouped aggregation stays above baseline through 20% symmetric flips. That test injects i.i.d. flips; a learned reward model's errors are correlated and biased toward particular failure modes, and we have not tested that noise model.

Sampleability. A behavior must be sampleable to be steered: if all rollouts fall on one side, no pair exists. This bounds ROAST to behaviors the model already produces unreliably, and is a limitation of *any* extract-from-behavior method (rates in App. C.1, decision rule in App. D).

Hyperparameters and scale. α is chosen by grid search on dev and performance degrades sharply if it is too large—the cliff and the boundary-optimum cases are documented in App. C.12. Gains are smallest on the weakest model we tested (Gemma3-1B), where noisy rollouts weaken the verifier partition, and on already-strong large models where headroom is limited.

Scope of the on-policy claim. Three separate restrictions apply to the word "on-policy," and we would rather over-state them than have the term carry weight it cannot.

It is a claim about the corpus, not about decoding. As §3.1 notes, a transformer's activations are fixed by the token sequence, so teacher-forcing a rollout the model generated yields exactly the activations that generating it yielded. Nothing in ROAST therefore depends on activations being collected *during* sampling, and no analogue of exposure bias is at work: what our method changes is the distribution the extraction texts are drawn from, and that is the entire mechanism.

The rollouts are filtered, so they are not a sample of p_{ar} either. They are model-induced and then verifier-partitioned, with questions whose rollouts do not split discarded outright (up to 86% of them, Table 11). The corpus we read from is thus a conditioned subsample of p_{ar}, not p_{ar} or the true task distribution.

The deployment corpus is a third distribution. At inference the model conditions on a prefix that is both self-generated *and steered*, which neither our extraction corpus nor the teacher-forced one matches. Closing that last gap would require extracting under the intervention itself—an iterative procedure we do not attempt. So the supported claim is comparative, that reading from the model's own outputs beats reading from supplied text, and not that rollouts are the ideal choice.

Comparisons not run. We compare against training-free contrastive directions only. Two omissions bound the conclusions: training on the same verified rollouts (Zelikman et al., 2022; Yuan et al., 2023; a different use of the same data) and a probe estimator on the same activations (Alain & Bengio, 2017; a different estimator on the same corpus). Neither is run, and Appendix B.7 states what each would have settled. On IFEval specifically, CAA and SADI have no defined construction—the task supplies no gold string to append—leaving RepE as the sole steering comparison on a column where we report a $+3.15$ margin, which thins that comparison.

Evaluation protocol. All numbers come from a single generate-then-extract harness (Appendix B.4) rather than per-benchmark official scripts, which makes methods comparable but leaves absolute values below published figures for several models. It also leaves open an alternative reading of part of our multiple-choice gains—improved answer extractability rather than improved selection—which a preliminary rescoring diagnostic (Table 9) makes concrete without resolving, since its configuration does not reproduce the margins it would need to explain. The generative benchmarks—GSM8K, MATH500, IFEval—do not pass through an option parser and are unaffected; they carry our largest margins and all of the trajectory evidence.