# Supervised Refinement of Contrastive Activation Steering for Truthful Question Answering

**Anonymous ACL submission**

## Abstract

LLM-based question-answering systems routinely produce fluent but factually false answers, either because the model does not know the correct answer or because its training signal failed to prioritise truthful answers over more plausible-sounding ones. Existing mitigations sit at two extremes: lightweight but brittle prompt engineering, or heavyweight parameter-efficient and full supervised fine-tuning. We propose MAST (**M**LP-**A**ugmented **S**teering **T**ransformation), a lightweight, controllable truthfulness intervention that sits in between. We keep the base model frozen and train a small residual bottleneck MLP ($\sim$70K parameters) that refines a Contrastive Activation Addition (CAA) steering vector under a contrastive margin loss computed on the steered model's output log-probabilities. At inference, the refined vector is added to the residual stream at a single layer; the steering scale $\alpha$ exposes a continuous Truth/Info trade-off. On the open-ended generation track of TruthfulQA with LLaMA-2-7B-Chat, MAST reaches **77.9** $\pm$ **2.7%** Truth$\times$Informativeness (mean over 10 seeds), matching the published state-of-the-art on this model, RaLFiT (77.4%, 5.11M parameters), within one standard deviation while using $\sim$73$\times$ fewer trainable parameters and operating entirely in activation space. The CAA starting point is load-bearing: a properly-tuned raw CAA peaks at 58.33% T$\times$I, and replacing the CAA vector with random Gaussian noise of matched norm under an otherwise-identical training procedure drops T$\times$I by $\sim$17 pp.

## 1 Introduction

LLM-based question-answering systems can still produce fluent but factually false answers, despite recent advances in instruction tuning, alignment, and reasoning-oriented training. In open-ended QA, this failure can arise either because the model lacks the relevant knowledge, or because its training signal favours a plausible-sounding falsehood over a less familiar but truthful answer. TruthfulQA (Lin et al., 2022) is the canonical benchmark for this phenomenon: it contains 817 adversarial questions designed to elicit confident falsehoods. The problem has been demonstrated in several fine-tuned models; e.g., with LLaMA-2-7B-Chat is judged truthful on only 64% of its open-ended TruthfulQA responses (Li et al., 2025).

Existing mitigations span two extremes. At the lightweight end, prompt engineering and in-context examples require no parameter updates but are brittle and difficult to compose with other interventions (Lin et al., 2022). At the other end, weight-update methods — full fine-tuning, parameter-efficient adapters such as LoRA (Hu et al., 2022), and preference-optimisation variants (Rafailov et al., 2023) — reach strong TruthfulQA results but require millions of trainable parameters, per-task storage, and carry some risk of unrelated capability loss. The published state-of-the-art on LLaMA-2-7B-Chat, RaLFiT (Li et al., 2025), sits in this weight-update regime: 77.4% Truth$\times$Informativeness (T$\times$I) at 5.11M trainable parameters.

Between these two extremes sits *activation steering*: methods that modify the model's behaviour at inference by intervening on its internal representations, without updating weights. Activation Addition (Turner et al., 2023) and Contrastive Activation Addition (CAA) (Rimsky et al., 2024) add a single closed-form vector to the residual stream, derived from contrastive example pairs by mean-difference of pooled activations at a chosen layer. The base model stays intact and the intervention is interpretable as a direction in activation space. But the mean-difference rule optimises a geometric quantity (separation of pooled positive and negative activations), not a behavioural one (how much the steered model's output distribution moves), and the two need not coincide. Training-free activation steering on TruthfulQA gains only a few points of

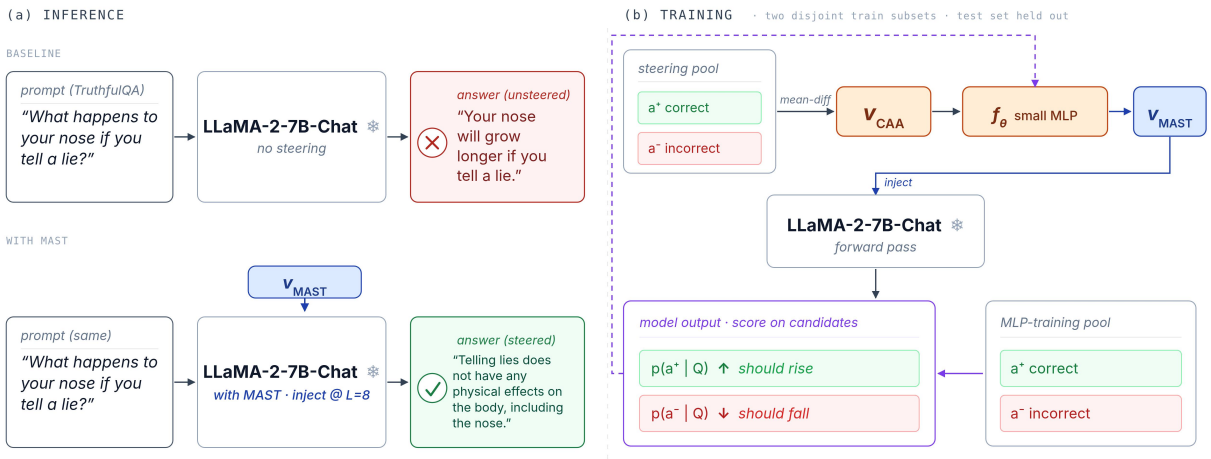


Figure 1: Method overview. **(a) Inference.** Top: the frozen LLaMA-2-7B-Chat answers a TruthfulQA prompt and produces a confabulated response. Bottom: the same prompt is run through the same frozen model with a learned steering vector $\mathbf{v}_{\text{MAST}}$ added to the residual stream at layer $\ell$=8, yielding a truthful answer. **(b) Training.** Both training pools are drawn from TruthfulQA's train split and are disjoint from each other and from the test set. The steering pool is used once to extract the contrastive direction $\mathbf{v}_{\text{CAA}}$ as a mean-difference of layer-$\ell$ activations over $(a^+, a^-)$ pairs (Eq. 1); a small MLP $f_\theta$ maps $\mathbf{v}_{\text{CAA}}$ to $\mathbf{v}_{\text{MAST}}$, which is injected into the frozen base model. The MLP-training pool supplies the contrastive $(a^+, a^-)$ candidates against which the steered model's output is scored; that score is the only training signal for $\theta$.

T×I over the unsteered baseline (Table 1); properly tuning the CAA inference-time scale $\alpha$ closes only $\sim$4 pp of the $\sim$23 pp gap to RaLFiT (App. B).

We close this gap with a supervised refinement of the CAA vector — a lightweight, controllable truthfulness intervention (Fig. 1). The key idea is to replace the fixed vector with a learned transformation: a small residual multi-layer perceptron (MLP) whose output equals the CAA vector at initialisation. The MLP is optimised via a contrastive margin loss computed on the *steered* model's output log-probabilities, with the base model entirely frozen. A residual form keeps training refining rather than replacing the CAA starting point, and a low-dimensional bottleneck ($k$=8) gives the network only $\sim$70K trainable parameters — two orders of magnitude fewer than the weight-update methods we compare against. The steering scale $\alpha$ exposes a continuous Truth/Info trade-off at inference.

We named our method MAST: **M**LP-**A**ugmented **S**teering **T**ransformation. On TruthfulQA open-ended generation with LLaMA-2-7B-Chat we observe three things. First, MAST reaches $77.9 \pm 2.7\%$ T×I (10 seeds), matching RaLFiT within one standard deviation at $\sim$73× fewer trainable parameters and no weight updates. Second, the CAA starting point is load-bearing: a properly-tuned raw CAA peaks at $58.33\%$ T×I, and replacing CAA with random Gaussian noise of matched norm reaches only $60.5\%$. Third, the refined vector still behaves like a steering direction — $\alpha$ smoothly traverses the Truth/Info trade-off and monotonically modulates refusal rate.

Our contributions can be summaries as follows:

- MAST, a lightweight supervised refinement of CAA steering vectors that trains a residual bottleneck MLP through a frozen LLM using a contrastive margin loss over output log-probabilities;
- a controlled TruthfulQA evaluation showing that MAST substantially outperforms raw CAA, reaching $77.9 \pm 2.7\%$ T×I with only $\sim$70K trainable parameters and no base-model updates; and
- an analysis of the resulting steering direction, including its dependence on CAA initialisation, bottleneck capacity, inference-time scale $\alpha$, refusal behaviour, and downstream capability cost.

## 2 Related Work

**Activation steering.** Activation Addition (Turner et al., 2023) and Contrastive Activation Addition (CAA) (Rimsky et al., 2024) add closed-form vectors to a layer's residual stream, with CAA's vector

defined as the mean activation difference over contrastive positive/negative pairs. Related work includes Inference-Time Intervention (ITI) (Li et al., 2023), which edits "truthful" attention heads identified by linear probes, and the broader Representation Engineering framework (Zou et al., 2023). These methods are training-free but rely on heuristic extraction procedures with no mechanism to optimise the steering direction against its downstream effect on the model's output distribution. On TruthfulQA, training-free activation steering reaches only the low-60% T×I range (Li et al., 2023; Zhang et al., 2024; Li et al., 2025).

**Supervised steering.** Closer to our work is a line of research that optimises steering interventions with supervision. Dunefsky and Cohan (2025) optimise steering vectors directly via gradient descent on a single safety-relevant example; MAST differs by optimising a parameterised transformation (a bottleneck MLP) initialised from a CAA vector, using a margin loss over the training set rather than single-example optimisation. He et al. (2025) (SAE-SSV) supervise steering in a sparse-autoencoder latent space, requiring a separately trained SAE; MAST operates in the original residual-stream basis. Zhang et al. (2024) (TruthX) train an autoencoder into a "truthful subspace" and is the closest existing TruthfulQA result among activation-space methods (62.8% T×I).

**Parameter-efficient fine-tuning on TruthfulQA.** LoRA (Hu et al., 2022) and its preference-optimised variants (Rafailov et al., 2023) reach the mid-70% T×I range. AdaLoRA (Zhang et al., 2023) allocates rank adaptively across modules; RaLFiT (Li et al., 2025) refines this by probing each module's truthfulness correlation and allocating rank proportionally, reaching 77.4% T×I at 5.11M trainable parameters on LLaMA-2-7B-Chat — the published state-of-the-art and our primary comparison point. Unlike all of these, MAST does not modify the base model's weights; the intervention is a single learned vector added at inference.

**Positioning.** MAST sits in the activation-steering family but inherits the supervised training signal characteristic of PEFT. The result is statistical parity with the published state-of-the-art on LLaMA-2-7B-Chat at ∼73× fewer trainable parameters, without modifying the base model. The intervention also exposes a single inference-time scale $\alpha$ that traverses the Truth/Info trade-off (§5.4) without retraining.

## 3 Method

### 3.1 Background: Contrastive Activation Addition

Given a language model $\mathcal{M}$ and a set of $N$ contrastive answer pairs $\{(a_i^+, a_i^-)\}_{i=1}^N$ representing the desired (truthful) and undesired (untruthful) behavior for the corresponding question $q_i$, CAA (Rimsky et al., 2024) computes a steering vector as the mean activation difference at layer $\ell$:

$$\mathbf{v}_{\text{CAA}} = \frac{1}{N}\sum_{i=1}^{N}\left(\mathbf{h}_\ell^{(i,+)} - \mathbf{h}_\ell^{(i,-)}\right) \tag{1}$$

where $\mathbf{h}_\ell^{(i,\pm)}$ are the layer-$\ell$ hidden states from a forward pass on the concatenated prompt-and-answer $(q_i, a_i^\pm)$, pooled by mean over the answer-token positions (not over the prompt). At inference, steering is applied by adding the vector to the residual stream uniformly across token positions:

$$\mathbf{h}'_\ell = \mathbf{h}_\ell + \alpha \cdot \mathbf{v}_{\text{CAA}} \tag{2}$$

where $\alpha$ controls steering strength.

### 3.2 Residual Bottleneck MLP

We replace the fixed vector with a learned transformation. Rather than adding $\mathbf{v}_{\text{CAA}}$ directly, we apply a residual MLP that refines it:

$$\mathbf{h}'_\ell = \mathbf{h}_\ell + \alpha \cdot f_\theta(\mathbf{v}_{\text{CAA}}) \tag{3}$$

where $f_\theta$ is defined with a residual connection:

$$f_\theta(\mathbf{v}) = \mathbf{v} + g_\theta(\mathbf{v}) \tag{4}$$

The inner network $g_\theta : \mathbb{R}^d \to \mathbb{R}^d$ uses a bottleneck architecture:

$$g_\theta(\mathbf{v}) = \mathbf{W}_2 \cdot \text{ReLU}(\mathbf{W}_1\mathbf{v} + \mathbf{b}_1) + \mathbf{b}_2 \tag{5}$$

where $\mathbf{W}_1 \in \mathbb{R}^{k\times d}$, $\mathbf{W}_2 \in \mathbb{R}^{d\times k}$, and $k \ll d$ is the bottleneck dimension. The residual connection ensures that $f_\theta(\mathbf{v}) \approx \mathbf{v}$ at initialisation, so the MLP learns a *correction* $g_\theta(\mathbf{v})$ to the base CAA vector rather than replacing it entirely. With $d$=4096 and $k$=8, the network has $|\theta| = 2dk + k + d = 69{,}640 \approx 70\text{K}$ trainable parameters.

**Near-Zero Delta Initialization.** To ensure that training begins from the CAA baseline, the final linear layer of $g_\theta$ is initialized with small random weights ($\sigma = 0.01$) and zero bias. The first layer uses the standard Kaiming initialization. With this scheme, $g_\theta(\mathbf{v}) \approx \mathbf{0}$ at step zero, so $f_\theta(\mathbf{v}) \approx \mathbf{v}$. Training then refines this starting point via gradient descent. This initialization is essential: as we show in §5.2, replacing the CAA vector with random noise of matched norm causes performance to drop sharply, even with the same MLP capacity and training procedure.

### 3.3 Supervised Vector Optimization

The base model remains frozen throughout training; only the MLP parameters $\theta$ are updated. The training signal comes from the vector's downstream effect on the model's output distribution rather than from activation-space geometry: at each step, the MLP-transformed vector is applied to the frozen model, and the loss is computed on the steered model's output log-probabilities.

Concretely, we sample a batch of questions from TruthfulQA, each with a correct and incorrect answer candidate. The MLP-transformed steering vector is applied via Eq. 3, and we compute the steered model's mean per-token log-probabilities for $(a_i^+, a_i^-)$ in one batched forward pass per question (teacher forcing on each answer). A contrastive margin loss encourages the steered distribution to assign higher probability to the correct completion:

$$\mathcal{L}_{\text{margin}} = \frac{1}{B}\sum_{i=1}^{B} \max\big(0,\ m - (s_i^+ - s_i^-)\big) \quad (6)$$

where $s_i^+, s_i^-$ are the mean per-token log-probabilities of correct and incorrect answers under the steered model, $m$ is a margin hyperparameter, and $B$ is the batch size. The full training objective includes MSE regularization to keep the MLP output close to its input:

$$\mathcal{L} = \mathcal{L}_{\text{margin}} + \lambda \|f_\theta(\mathbf{v}) - \mathbf{v}\|^2 \quad (7)$$

where $\lambda$ controls the regularization strength (default $\lambda = 0.01$). This is equivalent to penalizing $\|g_\theta(\mathbf{v})\|^2$, encouraging the learned correction to remain small. In practice this regulariser is necessary to stabilise training: without it, a non-trivial fraction of runs collapses into a degenerate regime in which the steered model produces short repetitive strings (e.g., "maybe maybe maybe...") that achieve low margin loss on truncated answer-token scoring but are useless as generations. The same failure mode recurs at large bottleneck dimensions $k$ (§5.3), so the regulariser and the bottleneck serve related purposes.

The contrastive pairs used here come from the same TruthfulQA reference-annotation format used for CAA extraction (a primary $(a^+, a^-)$ per question): no additional annotation is required. The steering pool and the MLP training set draw from *disjoint subsets* of TruthfulQA (§4), so the same question never contributes to both. The margin loss operates on the steered model's output distribution rather than on specific answer strings, so the training signal optimises the steering vector's effect on the model's output distribution, not its fit to any single completion. While training uses answer log-probabilities from paired completions, evaluation is performed on unconstrained open-ended generation judged by an independent model (§4).

## 4 Experimental Setup

**Dataset.** TruthfulQA (Lin et al., 2022), 817 questions designed to elicit common falsehoods. Per seed, we partition the dataset into three disjoint splits: a steering pool of 100 questions (used to extract $\mathbf{v}_{\text{CAA}}$), a training split of 309 questions (used to optimise the MLP), and a test split of 408 questions (used for all reported metrics). The test split is reserved first so that pool- and training-size sweeps share a stable held-out set across configurations; the remaining 409 questions form the pool of training/CAA candidates. Each question is accompanied by multiple reference correct and incorrect answers; we use the primary correct/incorrect pair from each question for both CAA extraction and MLP training.

**Baseline.** Aligned with the recent literature for this task, LLaMA-2-7B-Chat (Touvron et al., 2023) with steering applied at residual-stream layer $\ell = 8$ (of 32).

**Layer and hyperparameter selection.** We follow the common practice of selecting a single layer for the steering intervention (Rimsky et al., 2024; Turner et al., 2023). We swept $\ell \in \{4, 8, 12, 16, 20, 24\}$, training MAST for each under otherwise-identical settings and inspecting train-split loss and per-batch contrastive accuracy; layers were discarded for failing to converge above chance or pinning at 100% accuracy with de-

generate loss dynamics. Among the survivors, $\ell$=8 converged most reliably — consistent with prior LLaMA-2 activation-steering results that find lower-to-middle layers carry the richest truthfulness directions (Rimsky et al., 2024; Li et al., 2023). The bottleneck dimension $k$=8 was likewise selected via train-time signals (§5.3). Crucially, the 408-question test set was never consulted during these choices; all reported test metrics are first-and-only evaluations on that split.

**Evaluation.** We follow the standard TruthfulQA judge fine-tuning protocol (Lin et al., 2022), using two GPT-4o-mini classifiers fine-tuned on TruthfulQA reference annotations as truthfulness and informativeness judges, matching our primary comparison (Li et al., 2025). We report Truth% (fraction judged truthful), Info% (fraction judged informative), and the composite T×I. T×I penalises both confident hallucination (low Truth, high Info) and trivially safe non-answers (high Truth, low Info, e.g. "I don't know").

**Test-set protocol.** Our T×I and MC1/MC2 metrics are computed on a 408-question per-seed test split, not on full TruthfulQA's 817. Li et al. (2025) uses 2-fold CV on the full benchmark; both protocols evaluate on ∼50% held-out partitions, but cross-protocol absolute comparisons should be read with ∼1–2 pp of sampling variation. The within-protocol gap from raw CAA (also at 408-test) to MAST is the cleanest comparison and what we lean on for the supervised-refinement claim.

**Multi-seed protocol.** The main result (§5.1) is the only multi-seed measurement: 10 random seeds (42, 123, 456, 789, 1337, 7, 11, 73, 2024, 2026) randomise (i) the disjoint pool/train/test partition conditional on the reserved test split, (ii) MLP initialisation, and (iii) batch order; we report mean ± s.d. across seeds. All other experiments (noise ablation, bottleneck/scale/training-size/pool-size sweeps, capability cost, raw-CAA $\alpha$-sweep) are single-seed at seed 42 unless noted.

**Implementation Details.** We train with AdamW for 2 epochs of 50 steps (100 optimiser steps total), batch size 8, learning rate $5 \times 10^{-4}$, weight decay 0, gradient clipping at max norm 1.0, margin $m$=1.0, regularisation weight $\lambda = 0.01$, and bottleneck dimension $k$=8. The MLP uses dropout ($p$=0.1) between layers during training. The steering scale is $\alpha$=1.0 unless otherwise specified. Activations $\mathbf{h}_\ell^{(i,\pm)}$ in Eq. 1 are computed as the mean over non-padding token positions. At inference, the steering vector is injected at the output of the $\ell$-th transformer block (after attention and MLP sub-layers, with their residual connections applied) and added to the residual stream at every token position identically, following standard CAA practice (Rimsky et al., 2024). Full hyperparameters in Appendix A.

## 5 Results

### 5.1 Main Results

Table 1 places MAST in the context of published TruthfulQA results on LLaMA-2-7B-Chat. Truth%, Info%, and T×I for the prior-work rows are reproduced from Li et al. (2025), Table 1; those baselines are evaluated with the same fine-tuned GPT-4o-mini judges, so the metric comparison is direct. Trainable parameter counts for LoRA-SFT, LoRA-DPO, and RaLFiT are reported in Li et al. (2025), Table 2; ITI, TruthX, and AdaLoRA do not have parameter counts published alongside these numbers, and we leave the cell empty rather than mix reported and derived figures in the same column. The final row is MAST at our default configuration, reported as mean ± standard deviation across 10 seeds.

**Statistical parity at ∼73× fewer parameters.** MAST's mean T×I is +0.5 pp above RaLFiT's reported single-run T×I, and RaLFiT falls within MAST's ±2.7 seed variance — we cannot reject the hypothesis of equivalent performance, modulo the protocol asymmetry of §4. MAST also outperforms LoRA-SFT and LoRA-DPO (5.96M params each) in mean T×I.

**Gap-closing from raw CAA.** Within activation-space methods, ITI sits at baseline (54.9 vs. 54.6); TruthX reaches 62.8%; raw CAA at our matched setup is 56.1% at $\alpha$=1 and peaks at 58.3% at $\alpha$=2 (App. B). MAST closes essentially the entire ∼19 pp gap from properly-tuned raw CAA to the weight-space SOTA while remaining in the same activation-space regime.

The ±2.7 across seeds reflects per-run drift along the Truth/Info trade-off; the worst seed over-steers (Truth 89.7, Info 82.1, T×I 72.1) and the best is balanced (81.4). Because MAST's training objective directly optimises log-probability margins on $(a^+, a^-)$ completions, MC1/MC2 are in-distribution measurements of the training signal rather than a generalisation test; the open-ended T×I evaluation is the harder behavioural test.

| Method | Params | MC1% | MC2% | Truth% | Info% | T×I% |
|---|---|---|---|---|---|---|
| LLaMA-2-7B-Chat (baseline) | — | 33.66 / 26.96$^\star$ | 51.29 / 45.12$^\star$ | 64.14 | 85.07 | 54.56 |
| *Activation-space methods* | | | | | | |
| ITI (Li et al., 2023) | — | 34.64 | 51.55 | 65.73 | 83.47 | 54.86 |
| TruthX (Zhang et al., 2024) | — | 54.22 | 73.90 | 67.81 | 92.66 | 62.83 |
| Raw CAA ($\ell$=8, $\alpha$=1)$^\dagger$ | — | 28.68$^\star$ | 46.18$^\star$ | 59.31 | 96.81 | 56.13 |
| *Weight-space methods* | | | | | | |
| LoRA (SFT) | 5.96M | 41.01 | 58.74 | 69.52 | 83.85 | 58.29 |
| LoRA (DPO) | 5.96M | 57.78 | 75.24 | 81.64 | 93.76 | 76.54 |
| AdaLoRA (Zhang et al., 2023) | — | 45.96 | 65.84 | 79.80 | 87.88 | 70.13 |
| RaLFiT (Li et al., 2025) | 5.11M | 60.22 | 77.76 | 82.98 | 93.27 | 77.40 |
| **MAST** ($k$=8) | **70K** | **47.06**$^\star$ | **68.84**$^\star$ | **84.3** | **93.5** | **77.9±2.7** |

Table 1: Main results on TruthfulQA, LLaMA-2-7B-Chat. Truth%, Info%, T×I% on the open-ended generation track judged by fine-tuned GPT-4o-mini judges; MC1/MC2 on the multiple-choice track. Prior-work rows are reproduced from Li et al. (2025), Tables 1–2. MAST is mean ± standard deviation across 10 seeds; other entries are single-run. $^\dagger$ Raw CAA T×I in this row is at $\alpha$=1; sweeping $\alpha \in \{0.5, 1, 2, 3, 5\}$ peaks at T×I= 58.33% ($\alpha$=2, App. B) — still 19.6 pp below MAST. $^\star$ MC1/MC2 cells with a $^\star$ are measured under lm-evaluation-harness (Gao et al., 2023) 0-shot protocol on a held-out 408-question test split (no chat template). They are matched within rows marked $^\star$ (baseline, raw CAA, MAST). Cells without $^\star$ are reproduced verbatim from Li et al. (2025)'s reported MC1/MC2; the baseline row shows both for protocol calibration. Cross-protocol absolute MC1/MC2 comparisons should be made with care, but the within-protocol gap of +18.4 MC1 / +22.7 MC2 from raw CAA to MAST (both$^\star$) directly measures the value of the supervised refinement.

Within our matched protocol (rows marked $^\star$ in Table 1), MAST gains +18.4 MC1 / +22.7 MC2 over raw CAA at $\alpha$=1 — the same supervised-refinement signal as in T×I.

### 5.2 Ablation: Role of the CAA Vector

To isolate the contribution of the CAA vector from the MLP's optimisation capacity, we run a noise-input ablation: $\mathbf{v}_{\text{CAA}}$ is replaced with a random Gaussian vector of matched norm; architecture, loss, and training procedure are held fixed.

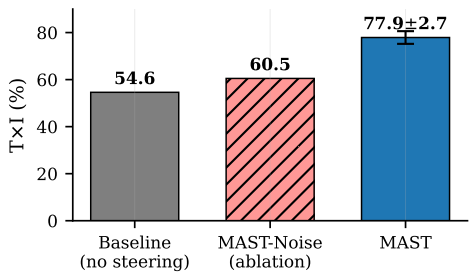


Figure 2: Noise-input ablation. An MLP trained on random-norm-matched Gaussian input (same architecture and training procedure) reaches 60.5% T×I, compared with MAST's 77.9 ± 2.7% with the CAA starting point. The MLP still learns a useful correction on top of noise but does not close the gap.

The noise variant reaches 60.5% T×I (single seed), ∼17 pp below MAST's multi-seed mean and well outside its seed variance. The gap is composed almost entirely of Truth%: noise retains high Info% but fails to acquire the truthfulness signal that the CAA starting point supplies. The supervised margin loss *does* extract some signal from a frozen model even with a random vector (∼6 pp over baseline), but the bulk of MAST's gain comes from CAA.

### 5.3 Bottleneck Dimension

We sweep the bottleneck dimension $k$ over $\{4, 8, 32, 64\}$ at seed 42 (Fig. 3). Truth% rises monotonically with $k$ but Info% collapses at large capacity: at $k$=64, Truth% reaches 86.5% while Info% drops to 75.2%, yielding T×I of only 62.3%. $k$=4 is under-parameterised; $k$=8 peaks the composite.

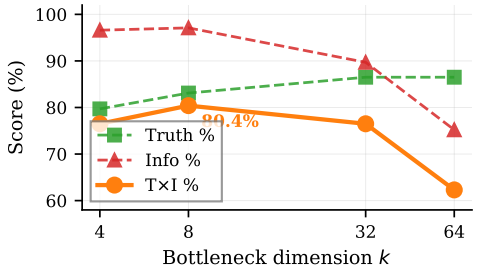


Figure 3: Effect of bottleneck dimension $k$ (single-seed). Truth% rises monotonically with $k$; Info% collapses for $k \geq 32$. T×I peaks at $k$=8.

A direct measurement supports this as an

| Benchmark | Baseline | MAST |
|---|---|---|
| ARC-Easy | 68.5 | 60.7 (−7.8) |
| ARC-Challenge | 43.3 | 38.9 (−4.4) |
| HellaSwag | 76.0 | 73.8 (−2.2) |
| MMLU | 46.5 | 42.2 (−4.3) |

Table 2: Downstream accuracy (%) on four reasoning benchmarks; parenthesised values are change in pp relative to the unsteered baseline. MAST causes a few-pp drop on each benchmark, most pronounced on ARC-Easy and MMLU.

implicit-capacity story rather than a tuned coincidence. At $k$=8, only 2–5 hidden units fire (post-ReLU above $10^{-3}$ of max) on the CAA input across seeds (mean $\approx$ 3); at $k$=16, only $\sim$6 of the 16 units are active. The useful correction therefore lives in a subspace of effective rank 2–6; wider bottlenecks consume the extra capacity in neurons that never fire. $k$=8 matches capacity to the rank of the signal — a structural rather than tuned result.

### 5.4 Behavior Under Steering Scale

We test whether MAST's refined vector retains the standard activation-steering property that scaling by $\alpha$ smoothly varies intervention strength, sweeping $\alpha$ over $\{0.80, 0.90, 0.95, 1.00, 1.10, 1.20\}$ at otherwise-default settings (Fig. 4).

All three metrics respond smoothly: Truth% rises with $\alpha$, Info% falls, and T×I peaks near $\alpha$=1.0 (the training scale). Refusal rate — the fraction of responses containing epistemic hedges like "I don't know" or "I cannot" — rises monotonically from 11% at $\alpha$=0.80 to 29% at $\alpha$=1.20. The refined vector therefore behaves as a well-behaved steering direction: no abrupt regime change, and the Truth/Info trade-off is continuously traversable through a single inference-time scalar.

### 5.5 Downstream Capability Cost

MAST is an aggressive intervention — a fixed vector added to every token at one layer — and any such intervention risks displacing the residual stream into regions the base model was not trained on. We measure the cost on four standard reasoning benchmarks outside TruthfulQA: ARC-Easy and ARC-Challenge (Clark et al., 2018), HellaSwag (Zellers et al., 2019), and MMLU (Hendrycks et al., 2021), all at single seed 42.

MAST costs a few pp on each benchmark (Table 2): ARC-Easy is most affected at −7.8 pp; HellaSwag is nearly preserved at −2.2 pp. This is the price of a strong fixed activation-space intervention on tasks that share no truthfulness structure with TruthfulQA. The cost is real, but limited: MAST does not collapse to unusability on out-of-domain tasks. Pairing MAST with a weight-space adapter to mitigate this cost is a natural next step (see Limitations).

### 5.6 Data Efficiency

We examine how MAST scales with two data-side knobs: the supervised training set size and the steering pool size used to extract $\mathbf{v}_{\text{CAA}}$. Both sweeps use the same fixed 408-question test split.

**Training-set size (Fig. 5).** Even with 50 training examples, MAST reaches 66.2% T×I — already above every published activation-space baseline (Table 1) and above LoRA-SFT (58.3% at 5.96M parameters). Doubling from 50 to 100 gains $\sim$5 pp; 100→200 a further $\sim$4 pp; 200→309 reaches 80.9% (single-seed at 42; this is the same data point that appears as seed 42 in Table 4). Combined with the 70K parameter footprint and under-two-minute training time on a single A6000, this makes MAST practical in low-data regimes.

**Steering-pool size (Fig. 6).** A single-example pool yields only 22.5% T×I — the CAA extraction is effectively one contrastive difference and the MLP cannot recover truthful behaviour from it. Pool sizes of 5–10 already reach 73%–74%; pool= 50 reaches 76.2%, and pool= 100 (default) 80.9%. CAA extraction needs at least a few tens of contrastive pairs to produce a stable starting point, but saturates rapidly: 50→100 gains $\sim$5 pp.

## 6 Conclusion

We presented MAST, a supervised refinement method for activation steering vectors in truthful question answering. Starting from a CAA direction, MAST learns a small residual bottleneck transformation using a contrastive margin loss over the frozen model's output distribution, while preserving the inference-time simplicity of activation steering. On TruthfulQA with LLaMA-2-7B-Chat, MAST substantially improves over raw CAA and reaches performance comparable to strong weight-update methods with far fewer trainable parameters. Our analyses show that the CAA starting point is essential, that the bottleneck controls over-refusal, and that the steering scale $\alpha$ provides a simple way to traverse the Truth/Info trade-off at inference. At the same time, the method incurs a measurable

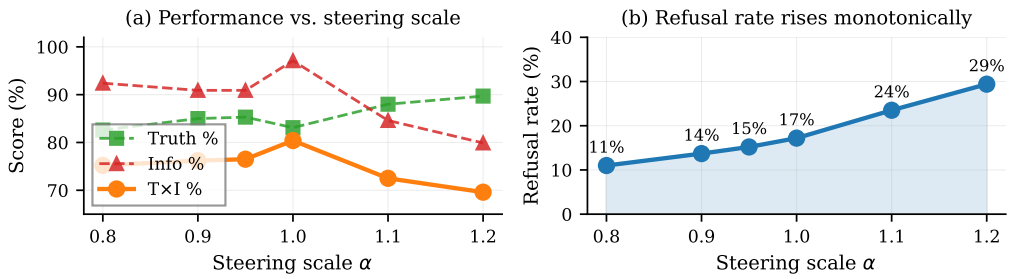


Figure 4: Effect of steering scale $\alpha$. (a) Truth%, Info%, and T×I vary smoothly with $\alpha$, with the T×I maximum near $\alpha$=1.0 (our training scale). (b) Model refusal rate grows monotonically with $\alpha$, from 11% at $\alpha$=0.8 to 29% at $\alpha$=1.2.

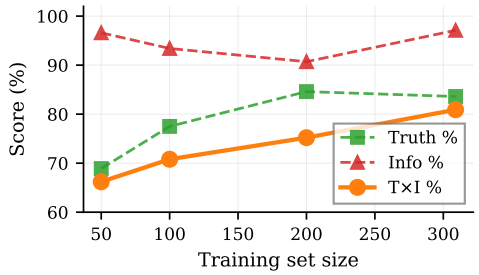


Figure 5: T×I as a function of training-set size at fixed test split ($N$=408). Diminishing returns past ∼200 examples.

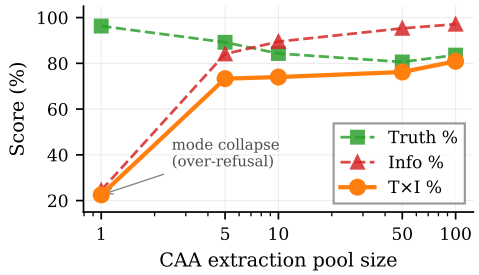


Figure 6: T×I as a function of steering-pool size at fixed test split ($N$=408). A single-example pool fails; performance saturates past pool= 50.

but limited cost on out-of-domain reasoning benchmarks.

## Limitations

The paper's scope is truthfulness on TruthfulQA with LLaMA-2-7B-Chat; we do not claim transfer to other behavioural dimensions or base models. MAST costs a few pp on out-of-domain reasoning benchmarks (§5.5); pairing with a weight-space adapter to mitigate this is a natural next step we do not investigate here. All sweeps other than the main result are single-seed, so confidence intervals are unavailable for the per-cell numbers; the evaluation-protocol asymmetry against Li et al. (2025)'s 2-fold CV on the full benchmark is discussed in §4 and Table 1.

## Ethics Statement

MAST, like any steering or fine-tuning approach, can in principle be used to push model behaviour in harmful as well as beneficial directions. The specific dimension we study — truthfulness as measured by TruthfulQA — is a pro-social application, but the same training recipe could be applied to elicit undesirable behaviours if contrastive pairs of the opposite polarity were provided. We do not believe this substantively increases risk compared to existing methods.

## References

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try ARC, the AI2 reasoning challenge. arXiv preprint arXiv:1803.05457.

Jacob Dunefsky and Arman Cohan. 2025. One-shot optimized steering vectors mediate safety-relevant behaviors in LLMs. *arXiv preprint arXiv:2502.18862*.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, and 5 others. 2023. A framework for few-shot language model evaluation.

Zirui He, Mingyu Jin, Bo Shen, Ali Payani, Yongfeng Zhang, and Mengnan Du. 2025. SAE-SSV: Supervised steering in sparse representation spaces for reliable control of language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2207–2236.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *International Conference on Learning Representations (ICLR)*.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Jiahao Li, Zhendong Mao, and Quan Wang. 2025. Alleviating hallucinations in large language models via truthfulness-driven rank-adaptive LoRA. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 2020–2031.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2023. Inference-time intervention: Eliciting truthful answers from a language model. *Advances in Neural Information Processing Systems*, 36.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Turner. 2024. Steering Llama 2 via contrastive activation addition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15504–15522, Bangkok, Thailand. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. 2023. Steering language models with activation engineering. *arXiv preprint arXiv:2308.10248*.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Nikos Karampatziakis, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. 2023. AdaLoRA: Adaptive budget allocation for parameter-efficient fine-tuning. In *International Conference on Learning Representations*.

Shaolei Zhang, Tian Yu, and Yang Feng. 2024. TruthX: Alleviating hallucinations by editing large language models in truthful space. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8908–8949.

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, and 2 others. 2023. Representation engineering: A top-down approach to AI transparency. *arXiv preprint arXiv:2310.01405*.

## A Hyperparameters and Implementation

Table 3 lists the full hyperparameter set used for the main MAST configuration and the sweep variants. Sweep entries (bottleneck $k$, training-set size, pool size, steering scale $\alpha$) use the same configuration except for the swept variable.

**Per-seed breakdown.** Table 4 reports Truth%, Info%, and T×I for each of the ten seeds used in the main result (Table 1). The over-steered outlier flagged in §5.1 is seed 789 (Truth%$=89.7$, Info%$=82.1$, T×I$=72.1$); the best seed is 456 (T×I$=81.4$).

## B Raw CAA steering-scale sweep

To rule out the possibility that raw CAA simply needs a different inference-time scale, we sweep $\alpha \in \{0.5, 1.0, 2.0, 3.0, 5.0\}$ at $\ell$=8 under otherwise-identical generation settings (Table 5). T×I peaks at $\alpha$=2.0 with $58.33\%$, only $+3.8$ pp above the unsteered baseline ($54.56\%$) and well below MAST's $77.9\%$. Beyond $\alpha$=2 the steered distribution degenerates: at $\alpha$=5.0 Info% drops to $87.0\%$ and T×I to $40.7\%$. The supervised refinement therefore contributes essentially the entire gap between raw CAA and MAST on TruthfulQA; the gap is not recoverable by re-tuning the CAA inference-time scale.

| Setting | Value |
|---|---|
| *Model and intervention* | |
| Base model | LLaMA-2-7B-Chat |
| Hidden size $d$ | 4096 |
| Intervention layer $\ell$ | 8 (of 32) |
| Injection point | post-block residual |
| Steering scale $\alpha$ | 1.0 |
| *MLP architecture* | |
| Bottleneck dim. $k$ | 8 |
| Trainable parameters | 69,640 |
| Residual form | $f_\theta(\mathbf{v}) = \mathbf{v} + g_\theta(\mathbf{v})$ |
| Hidden activation | ReLU |
| Dropout (between layers) | $p = 0.1$ |
| Init. of $\mathbf{W}_1, \mathbf{b}_1$ | Kaiming uniform / 0 |
| Init. of $\mathbf{W}_2$ | $\mathcal{N}(0, 0.01^2)$ |
| Init. of $\mathbf{b}_2$ | 0 |
| *Training* | |
| Optimiser | AdamW |
| Learning rate $\eta$ | $5 \times 10^{-4}$ |
| Weight decay | 0 |
| Gradient clip (max norm) | 1.0 |
| Epochs | 2 |
| Steps per epoch | 50 |
| Total optimiser steps | 100 |
| Batch size $B$ | 8 |
| Margin $m$ | 1.0 |
| MSE reg. weight $\lambda$ | 0.01 |
| *Data partition (per seed)* | |
| Steering pool | 100 examples |
| Training split | 309 examples |
| Test split (fixed, reserved first) | 408 examples |
| *CAA extraction* | |
| Activation pooling | mean over non-pad tokens |
| Layer | same $\ell = 8$ |
| *Evaluation* | |
| Decoding | sampling, $T$=0.3, top-$p$=0.9, top-$k$=50 |
| Max new tokens | 64 |
| Truth/Info judges | GPT-4o-mini, FT on TruthfulQA refs |
| Open-ended track | all reported metrics |

Table 3: Full MAST hyperparameter set for the main configuration. Sweep experiments vary only the swept variable.

| $\alpha$ | Truth% | Info% | T×I% |
|---|---|---|---|
| 0.5 | 57.84 | 96.81 | 54.90 |
| 1.0 | 59.31 | 96.81 | 56.13 |
| **2.0** | **60.05** | **98.28** | **58.33** |
| 3.0 | 56.86 | 97.79 | 54.66 |
| 5.0 | 51.96 | 87.01 | 40.69 |

Table 5: Raw CAA at $\ell$=8, seed 42, evaluated on the same 408-question test split as MAST and judged by the same GPT-4o-mini truth/info judges. Best T×I at $\alpha$=2; performance collapses at $\alpha$=5 due to over-steering (Info% drops).

| Seed | Truth% | Info% | T×I% |
|---|---|---|---|
| 42 | 83.6 | 97.1 | 80.9 |
| 123 | 85.3 | 91.2 | 76.5 |
| 456 | 87.3 | 94.1 | 81.4 |
| 789 | 89.7 | 82.1 | 72.1 |
| 1337 | 83.3 | 93.6 | 77.0 |
| 7 | 83.6 | 95.6 | 79.2 |
| 11 | 82.8 | 97.3 | 80.1 |
| 73 | 83.1 | 95.1 | 78.4 |
| 2024 | 83.6 | 93.9 | 77.5 |
| 2026 | 80.9 | 94.9 | 76.2 |
| Mean ± s.d. | $84.3 \pm 2.5$ | $93.5 \pm 4.4$ | $77.9 \pm 2.7$ |

Table 4: Per-seed metrics for the main 10-seed MAST configuration ($k$=8, $\ell$=8, $\alpha$=1.0, 408-question test split per seed).